

RN SHETTY TRUST ®



RNS INSTITUTE OF TECHNOLOGY

Autonomous Institution affiliated to Visvesvaraya Technological University, Belagavi

Approved By AICTE, New Delhi. Accredited by NAAC 'A+' Grade

RR Nagar, Dr. Vishnuvardhan Road, Bengaluru - 560 098

Ph:(080)28611880,28611881 URL: www.rnsit.ac.in



25
RNSIT

DEPARTMENT OF COMPUTER SCIENCE
& ENGINEERING

BACHELOR OF ENGINEERING

Scheme and Syllabus of V - VIII Semesters

2024 SCHEME



🌐 www.rnsit.ac.in

Scheme and Syllabus Batch 2024 – 2028

(As per NEP-2020)

(Effective from the academic year 2024 – 25)

An Institute with a Difference

About RNSIT

RNS Institute of Technology (RNSIT), established in 2001, was founded by Dr. R. N. Shetty, a renowned industrialist and philanthropist, with the vision of imparting quality education in Engineering, Technology, and Management.

RNSIT is one of the premier engineering institutions in the country and is steadily advancing towards a global presence through excellence in academics and placements.

The institute features a lush green campus, state-of-the-art infrastructure, well-equipped laboratories, and a vibrant academic environment. With numerous initiatives, consistent VTU ranks, outstanding placements, cultural events, and sports activities, RNSIT has become a preferred destination for engineering aspirants. The institution hosts over 4000 students and more than 300 faculty members.

Vision

Building RNSIT into a World Class Institution

Mission

To impart high-quality education in Engineering, Technology, and Management with a difference, enabling students to excel in their careers by:

- Attracting quality students and preparing them with a strong foundation in fundamentals to achieve excellence in various walks of life.
- Imparting value-based, need-based, choice-based, and skill-based professional education to shape disciplined, world-class professionals with social responsibility.
- Promoting excellence in teaching, research, and consultancy to enhance academic engagement among faculty and students.
- Exposing students to emerging frontiers of knowledge, preparing them for industry, entrepreneurship, higher studies, and research & development.
- Providing freedom of action and choice for all stakeholders with improved visibility and opportunities.

About Computer Science and Engineering

The Department of Computer Science & Engineering (CSE) at RNS Institute of Technology (RNSIT) is where academic excellence and innovation converge. For over 24 years, the department has proudly shaped future engineers by providing quality education, cutting-edge research opportunities, and a strong foundation for successful careers.

Accredited by the NBA, the department is committed to maintaining state-of-the-art laboratories and strong industry partnerships, ensuring a holistic learning experience. As an Autonomous Institution affiliated to VTU, the department offers an undergraduate program (established in 2001) with an intake of 720 students and houses a VTU-recognized R&D center (established in 2009).

Currently, 52 scholars are pursuing research, and 30 scholars have been awarded their degrees. The department has established Centers of Excellence in Machine Learning, FitterLife, and Agile Labs to advance in specialized fields. It has organized over 50 FDPs, 4 International Conferences, 3 National Conferences, and 400+ invited talks, benefiting both faculty and students.

The department strives to realize the vision of the institution by promoting excellence in teaching and providing a strong foundation in the fundamentals of computer science.

Vision

Preparing Better Computer Professionals for a Real World

Mission

- Imparting solid foundations and applied aspects in both Computer Science theory and programming practices.
- Providing training and encouraging R&D and consultancy services in frontier areas of Computer Science and Engineering with a global outlook.
- Fostering the highest ideals of ethics, values, and creating awareness of the role of computing in the global environment.
- Educating and preparing graduates who are highly sought after, productive, and well-respected for their work culture.
- Supporting and encouraging lifelong learning.

Program Educational Objectives (PEO)

- PEO1:** Graduates will build successful careers in computing by designing, analyzing, implementing, and evaluating reliable hardware/software systems to solve real-world problems.
- PEO2:** Graduates will practice engineering with ethics, effective communication, teamwork, leadership, and entrepreneurial skills while contributing responsibly to organizations and society.
- PEO3:** Graduates will pursue higher education, research, certifications, and continuous professional development to adapt to evolving technologies.

Program Outcomes (PO)

- PO1: Engineering Knowledge:** Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.
- PO2: Problem Analysis:** Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)
- PO3: Design/Development of Solutions:** Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)
- PO4: Conduct Investigations of Complex Problems:** Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis and interpretation of data to provide valid conclusions. (WK8)
- PO5: Engineering Tool Usage:** Create, select and apply appropriate techniques, resources and modern engineering and IT tools, including prediction and modelling, recognizing their limitations to solve complex engineering problems. (WK2 and WK6)
- PO6: The Engineer and The World:** Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5 and WK7)

- PO7: Ethics:** Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national and international laws. (WK9)
- PO8: Individual and Collaborative Teamwork:** Function effectively as an individual, and as a member or leader in diverse and multidisciplinary teams.
- PO9: Communication:** Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, and make effective presentations considering cultural, language and learning differences.
- PO10: Project Management and Finance:** Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects in multidisciplinary environments.
- PO11: Life-Long Learning:** Recognize the need for and have the preparation and ability for (i) independent and life-long learning, (ii) adaptability to new and emerging technologies, and (iii) critical thinking in the broadest context of technological change. (WK8)

Program Specific Outcomes (PSO)

- PSO1:** Apply core computer science principles to design and develop hardware and software solutions with effective communication and teamwork.
- PSO2:** Develop innovative, SDG-aligned solutions using emerging technologies with research and lifelong learning.



B.E. in Computer Science and Engineering: Autonomous Scheme and Syllabus
5th Semester

SL NO	COURSE	COURSE CODE	COURSE TITLE	TD/PSB	TEACHING HOURS PER WEEK			(TW + SL) /Sem	EXAMINATION				CREDITS
					THEORY LECTURE (L)	TUTORIAL (T)	PRACTICAL (P)		DURATION IN HOURS	CIE MARKS	SEE MARKS	TOTAL MARKS	
1	HSMS	BCS501	SOFTWARE ENGINEERING AND PROJECT MANAGEMENT	CSE	3	2	0	45	3	50	50	100	4
2	IPCC	BCS502	COMPUTER NETWORKS	CSE	3	0	2	45	3	50	50	100	4
3	IPCC	BCS503	THEORY OF COMPUTATION AND COMPILER DESIGN	CSE	3	0	2	45	3	50	50	100	4
4	PCCL	BCSL504	FULL STACK DEVELOPMENT	CSE	0	0	2	0	3	50	50	100	1
5	PEC	BCS515X	PROFESSIONAL ELECTIVE (BUCKET LIST E)	CSE	3	0	0	45	3	50	50	100	3
6	PROJ	BCS586	MINI PROJECT	CSE	0	0	2	30	3	100	0	100	2
7	BSC	BENS507/ BBOE507	ENVIRONMENTAL STUDIES / BIOLOGY FOR ENGINEERS	DEPT	2	0	0	30	2	50	50	100	2
8	AEC	BRMK557	RESEARCH METHODOLOGY AND IPR	ANY	2	0	0	30	2	50	50	100	2
9	MC	BNSK559	NATIONAL SERVICE SCHEME	ANY	0	0	2	0	–	100	0	100	0
		BPEK559	PHYSICAL EDUCATION										
		BYOK559	YOGA										
TOTAL										550	350	900	22

NOTE: 1st SEMESTER CHEMISTRY CYCLE SECTIONS WILL STUDY “ENVIRONMENTAL STUDIES” IN 4th SEMESTER AND “BIOLOGY FOR ENGINEERS” IN 5th SEMESTER

NOTE: 1st SEMESTER PHYSICS CYCLE SECTIONS WILL STUDY “BIOLOGY FOR ENGINEERS” IN 4th SEMESTER AND “ENVIRONMENTAL STUDIES” IN 5th SEMESTER



RN Shetty Trust®

RNS INSTITUTE OF
TECHNOLOGY

An Autonomous Institution, Affiliated to VTU,
Accredited with NAAC A+ Grade

Department of Computer Science
and Engineering

Autonomous Scheme (Effective from Academic
Year 2024 – 25)



BUCKET LIST E — PROFESSIONAL ELECTIVE BCS515X	
BCS515A	ADVANCED ALGORITHMS
BCS515B	BUSINESS INTELLIGENCE
BCS515C	UNIX SYSTEM PROGRAMMING
BCS515D	HADOOP AND SPARK
BCS515E	ETL AND DATA WAREHOUSING
BCS515F	EDGE AI AND TINY ML
BCS515G	DATA GOVERNANCE AND SECURITY
BCS515H	SITE RELIABILITY ENGINEERING
BCS515I	API DEVELOPMENT AND MICROSERVICES
BCS515J	INTRODUCTION TO COMPUTER FORENSICS



B.E. in Computer Science and Engineering : Autonomous Scheme and Syllabus
6th Semester

SL NO	COURSE	COURSE CODE	COURSE TITLE	TD/PSB	TEACHING HOURS PER WEEK			(TW + SL) /Sem	EXAMINATION				CREDITS
					THEORY LECTURE (L)	TUTORIAL (T)	PRACTICAL (P)		DURATION IN HOURS	CIE MARKS	SEE MARKS	TOTAL MARKS	
1	IPCC	BCS601	CLOUD COMPUTING	CSE	3	0	2	45	3	50	50	100	4
2	PCC	BCS602	ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING	CSE	3	2	0	45	3	50	50	100	4
3	PEC	BCS613X	PROFESSIONAL ELECTIVE (BUCKET LIST F)	CSE	3	0	0	45	3	50	50	100	3
4	OEC	BXX654X	OPEN ELECTIVE (BUCKET LIST G)	CSE AND ALLIED	3	0	0	45	3	50	50	100	3
5	PROJ	BCS685	MAJOR PROJECT PHASE-1	CSE	0	0	4	0	3	100	0	100	2
6	PCCL	BCSL606	ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING LABORATORY	CSE	0	0	2	0	3	50	50	100	1
7	AEC/SEC	BCST/L657X	(BUCKET LIST H)	CSE	IF COURSE IS THEORY				2	50	50	100	1
					1	0	0	15					
					IF COURSE IS PRACTICAL				3				
					0	0	2	0					
8	MC	BIKS608	INDIAN KNOWLEDGE SYSTEM	ANY	1	0	0	0	2	100	–	100	0
9	MC	BNSK559	NATIONAL SERVICE SCHEME	ANY	0	0	2	0	2	100	0	100	0
		BPEK559	PHYSICAL EDUCATION										
		BYOK559	YOGA										
TOTAL										600	300	900	18



ESTD: 2001
An Institute with a Difference

RN Shetty Trust®

RNS INSTITUTE OF
TECHNOLOGY

An Autonomous Institution, Affiliated to VTU,
Accredited with NAAC A+ Grade

Department of Computer Science
and Engineering

Autonomous Scheme (Effective from Academic
Year 2024 – 25)



BUCKET LIST F — PROFESSIONAL ELECTIVE BCS612X

BCS613A	QUANTUM COMPUTING
BCS613B	MULTI-AGENT SYSTEMS
BCS613C	AUGMENTED REALITY AND VIRTUAL REALITY
BCS613D	CRYPTOGRAPHY AND NETWORK SECURITY
BCS613E	SECURE PROGRAMMING
BCS613F	DESIGN PATTERNS AND CLEAN CODE
BCS613G	APPLIED STATISTICAL MODELLING
BCS613H	PUBLIC CLOUD ARCHITECTURE
BCS613I	INTRODUCTION TO DATA CENTER NETWORKING
BCS613J	AI DRIVEN NETWORKS
BCS613K	DEVOPS AND CI/CD PIPELINES

BUCKET LIST G — OPEN ELECTIVE BXX654X

BCS654A	INTRODUCTION TO DATA STRUCTURES
BCY654B	FUNDAMENTALS OF OPERATING SYSTEMS
BCI654C	INTRODUCTION TO DBMS
BCD654D	FOUNDATIONS OF DATA SCIENCE



RN Shetty Trust®

RNS INSTITUTE OF
TECHNOLOGY

An Autonomous Institution, Affiliated to VTU,
Accredited with NAAC A+ Grade

Department of Computer Science
and Engineering

Autonomous Scheme (Effective from Academic
Year 2024 – 25)



BUCKET LIST H — AEC BCSL657X	
BCSL657A	AUTOMATED SOFTWARE TESTING
BCSL657B	ETHICAL HACKING AND VULNERABILITY ASSESSMENT
BCSL657C	MLOPS
BCSL657D	SYSTEM DESIGN AND SCALABLE ARCHITECTURE
BCSL657E	COMPETITIVE PROGRAMMING
BCSL657F	MONITORING AND LOGGING SYSTEMS
BCSL657G	REINFORCEMENT LEARNING
BCSL657H	TIME SERIES ANALYSIS
BCSL657I	SERVERLESS COMPUTING
BCSL657J	SCALA AND JULIA



B.E. in Computer Science and Engineering : Autonomous Scheme and Syllabus
7th Semester

SL NO	COURSE	COURSE CODE	COURSE TITLE	TD/PSB	TEACHING HOURS PER WEEK			(TW + SL) /Sem	EXAMINATION				CREDITS
					THEORY LECTURE (L)	TUTORIAL (T)	PRACTICAL (P)		DURATION IN HOURS	CIE MARKS	SEE MARKS	TOTAL MARKS	
1	IPCC	BCS701	HIGH PERFORMANCE COMPUTING	CSE	3	0	2	45	3	50	50	100	4
2	IPCC	BCS702	DATA ENGINEERING	CSE	3	0	2	45	3	50	50	100	4
3	PCC	BCS703	BLOCKCHAIN TECHNOLOGY	CSE	3	2	0	45	3	50	50	100	4
4	PEC	BCS714X	PROFESSIONAL ELECTIVE (BUCKET LIST I)	CSE	3	0	0	45	3	50	50	100	3
5	OEC	BXX755X	OPEN ELECTIVE (BUCKET LIST J)	CSE AND ALLIED	3	0	0	45	3	50	50	100	3
6	PROJ	BCS786	MAJOR PROJECT PHASE-2	CSE	0	0	12	0	3	100	100	200	6
TOTAL										350	350	700	24



ESTD: 2001
An Institute with a Difference

RN Shetty Trust®

RNS INSTITUTE OF
TECHNOLOGY

An Autonomous Institution, Affiliated to VTU,
Accredited with NAAC A+ Grade

Department of Computer Science
and Engineering

Autonomous Scheme (Effective from Academic
Year 2024 – 25)



BUCKET LIST I — PROFESSIONAL ELECTIVE BCS714X

BCS714A	EXPLAINABLE SOFT COMPUTING
BCS714B	DEEP LEARNING
BCS714C	COMPUTER VISION AND IMAGE PROCESSING
BCS714D	NATURAL LANGUAGE PROCESSING AND GEN AI
BCS714E	CLOUD SECURITY AND ZERO TRUST ARCHITECTURE
BCS714F	NETWORK DESIGN FUNDAMENTALS
BCS714G	UI/UX DESIGN FUNDAMENTALS
BCS714H	CLOUD PROFESSIONAL MACHINE LEARNING ENGINEER

BUCKET LIST J — OPEN ELECTIVE BXX755X

BCS755A	INTRODUCTION TO ALGORITHMS
BCY755B	FOUNDATIONS OF INFORMATION SECURITY
BCD755C	FULL STACK WEB DEVELOPMENT
BCI755D	INTRODUCTION TO MACHINE LEARNING



B.E. in Computer Science and Engineering : Autonomous Scheme and Syllabus
8th Semester

SL NO	COURSE	COURSE CODE	COURSE TITLE	TD/PSB	TEACHING HOURS PER WEEK			(TW + SL) /Sem	EXAMINATION				CREDITS
					THEORY LECTURE (L)	TUTORIAL (T)	PRACTICAL (P)		DURATION IN HOURS	CIE MARKS	SEE MARKS	TOTAL MARKS	
1	PEC	BCS811X	PROFESSIONAL ELECTIVE (BUCKET LIST K) (ONLINE)	ONLINE	3	0	0	45	3	50	50	100	3
2	OEC	BXX852X	OPEN ELECTIVE (BUCKET LIST L) (ONLINE)	ONLINE	3	0	0	45	3	50	50	100	3
3	INT	BCS803	INTERNSHIP (INDUSTRY/RESEARCH - MINIMUM 14 WEEKS)	CSE	0	0	20	0	3	100	100	200	10
TOTAL										200	200	400	16

BUCKET LIST K — PROFESSIONAL ELECTIVE BCS811X (ONLINE COURSE)

BCS811X	BOS WILL PUBLISH COURSES BASED ON THE AVAILABILITY
---------	--

BUCKET LIST L — OPEN ELECTIVE BXX852X (ONLINE COURSE)

BXX852X	BOS WILL PUBLISH COURSES BASED ON THE AVAILABILITY
---------	--



B.E. in Computer Science and Engineering : Autonomous Scheme and Syllabus

7th Semester

SCHEME - A - VII SEMESTER (Swappable VII and VIII SEMESTER)

SL NO	COURSE	COURSE CODE	COURSE TITLE	TD/PSB	TEACHING HOURS PER WEEK			(TW + SL) /Sem	EXAMINATION				CREDITS
					THEORY LECTURE (L)	TUTORIAL (T)	PRACTICAL (P)		DURATION IN HOURS	CIE MARKS	SEE MARKS	TOTAL MARKS	
1	IPCC	BCS701	HIGH PERFORMANCE COMPUTING	CSE	3	0	2	45	3	50	50	100	4
2	IPCC	BCS702	DATA ENGINEERING	CSE	3	0	2	45	3	50	50	100	4
3	PCC	BCS703	BLOCKCHAIN TECHNOLOGY	CSE	3	2	0	45	3	50	50	100	4
4	PEC	BCS714X	PROFESSIONAL ELECTIVE (BUCKET LIST I)	CSE	3	0	0	45	3	50	50	100	3
5	OEC	BXX755X	OPEN ELECTIVE (BUCKET LIST J)	CSE AND ALLIED	3	0	0	45	3	50	50	100	3
6	PROJ	BCS786	MAJOR PROJECT PHASE-2	CSE	0	0	12	0	3	100	100	200	6
TOTAL										350	350	700	24

BUCKET LIST I — PROFESSIONAL ELECTIVE BCS714X

BCS714A	EXPLAINABLE SOFT COMPUTING
BCS714B	DEEP LEARNING
BCS714C	COMPUTER VISION AND IMAGE PROCESSING
BCS714D	NATURAL LANGUAGE PROCESSING AND GEN AI
BCS714E	CLOUD SECURITY AND ZERO TRUST ARCHITECTURE
BCS714F	NETWORK DESIGN FUNDAMENTALS
BCS714G	UI/UX DESIGN FUNDAMENTALS
BCS714H	CLOUD PROFESSIONAL MACHINE LEARNING ENGINEER

BUCKET LIST J — OPEN ELECTIVE BXX755X

BCS755A	INTRODUCTION TO ALGORITHMS
BCY755B	FOUNDATIONS OF INFORMATION SECURITY
BCD755C	FULL STACK WEB DEVELOPMENT
BCI755D	INTRODUCTION TO MACHINE LEARNING

Abbreviations: PCC: Professional Core Course, PCCL: Professional Core Course Laboratory, PEC: Professional Elective Course, OEC: Open Elective Course, PR: Project Work, L: Lecture, T: Tutorial, P: Practical, SDA: Skill Development Activity, CIE: Continuous Internal Evaluation, SEE: Semester End Evaluation, TD: Teaching Department, PSB: Paper Setting Department, PROJ: Project Work.

Note: VII and VIII Semesters of IV Year of the Program

1. Institutions can swap the VII and VIII semester Schemes of Teaching and Examinations to accommodate research internships/industry internships after the VI semester.
2. Credits earned for the courses of VII and VIII semester Schemes of Teaching and Examinations shall be counted against the corresponding semesters, whether the VII or VIII semester is completed during the beginning of the IV year or the later part of the IV year of the program.



B.E. in Computer Science and Engineering : Autonomous Scheme and Syllabus

8th Semester

SCHEME - A - VIII SEMESTER (Swappable VII and VIII SEMESTER)

SL NO	COURSE	COURSE CODE	COURSE TITLE	TD/PSB	TEACHING HOURS PER WEEK			(TW + SL) /Sem	EXAMINATION				CREDITS
					THEORY LECTURE (L)	TUTORIAL (T)	PRACTICAL (P)		DURATION IN HOURS	CIE MARKS	SEE MARKS	TOTAL MARKS	
1	PEC	BCS811X	PROFESSIONAL ELECTIVE (BUCKET LIST K) (ONLINE)	ONLINE	3	0	0	45	3	50	50	100	3
2	OEC	BXX852X	OPEN ELECTIVE (BUCKET LIST L) (ONLINE)	ONLINE	3	0	0	45	3	50	50	100	3
3	INT	BCS803	INTERNSHIP (INDUSTRY/RESEARCH - MINIMUM 14 WEEKS)	CSE	0	0	20	0	3	100	100	200	10
TOTAL										200	200	400	16

BUCKET LIST K — PROFESSIONAL ELECTIVE BCS811X (ONLINE COURSE)

BCS811X	BOS WILL PUBLISH COURSES BASED ON THE AVAILABILITY
---------	--

BUCKET LIST L — OPEN ELECTIVE BXX852X (ONLINE COURSE)

BXX852X	BOS WILL PUBLISH COURSES BASED ON THE AVAILABILITY
---------	--



B.E. in Computer Science and Engineering : Autonomous Scheme and Syllabus

6th Semester

Scheme - B - VI SEMESTER for the candidates who seek a two - semester internship with project work/Start - up

SL NO	COURSE	COURSE CODE	COURSE TITLE	TD/PSB	TEACHING HOURS PER WEEK			(TW + SL) /Sem	EXAMINATION				CREDITS
					THEORY LECTURE (L)	TUTORIAL (T)	PRACTICAL (P)		DURATION IN HOURS	CIE MARKS	SEE MARKS	TOTAL MARKS	
1	IPCC	BCS601	CLOUD COMPUTING	CSE	3	0	2	45	3	50	50	100	4
2	PCC	BCS602	ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING	CSE	3	2	0	45	3	50	50	100	4
3	PEC	BCS613X	PROFESSIONAL ELECTIVE (BUCKET LIST F)	CSE	3	0	0	45	3	50	50	100	3
4	OEC	BXX654X	OPEN ELECTIVE (BUCKET LIST G)	CSE AND ALLIED	3	0	0	45	3	50	50	100	3
5	PCCL	BCSL606	ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING LABORATORY	CSE	0	0	2	0	3	50	50	100	1
6	AEC/SEC	BCST/L657X	(BUCKET LIST H)	CSE	IF COURSE IS THEORY			2	50	50	100	1	
					1	0	0						15
					IF COURSE IS PRACTICAL			3					
					0	0	2						0
7	MC	BIKS608	INDIAN KNOWLEDGE SYSTEM	ANY	1	0	0	0	2	100	–	100	0
8	MC	BNSK659	NATIONAL SERVICE SCHEME	ANY	0	0	2	0	2	100	–	100	0
		BPEK659	PHYSICAL EDUCATION										
		BYOK659	YOGA										
TOTAL										600	300	900	16



RN Shetty Trust®

RNS INSTITUTE OF TECHNOLOGY

An Autonomous Institution, Affiliated to VTU,
Accredited with NAAC A+ Grade

Department of Computer Science
and Engineering

Autonomous Scheme (Effective from Academic
Year 2024 – 25)



BUCKET LIST F — PROFESSIONAL ELECTIVE BCS612X	
BCS613A	QUANTUM COMPUTING
BCS613B	MULTIAGENT SYSTEMS
BCS613C	AUGMENTED REALITY AND VIRTUAL REALITY
BCS613D	CRYPTOGRAPHY AND NETWORK SECURITY
BCS613E	SECURE PROGRAMMING
BCS613F	DESIGN PATTERNS AND CLEAN CODE
BCS613G	APPLIED STATISTICAL MODELLING
BCS613H	PUBLIC CLOUD ARCHITECTURE
BCS613I	INTRODUCTION TO DATA CENTER NETWORKING
BCS613J	AI DRIVEN NETWORKS
BCS613K	DEVOPS AND CI/CD PIPELINES

BUCKET LIST G — OPEN ELECTIVE BXX654X	
BCS654A	INTRODUCTION TO DATA STRUCTURES
BCY654B	FUNDAMENTALS OF OPERATING SYSTEMS
BCI654C	INTRODUCTION TO DBMS
BCD654D	FOUNDATIONS OF DATA SCIENCE



RN Shetty Trust®

RNS INSTITUTE OF
TECHNOLOGY

An Autonomous Institution, Affiliated to VTU,
Accredited with NAAC A+ Grade

Department of Computer Science
and Engineering

Autonomous Scheme (Effective from Academic
Year 2024 – 25)



BUCKET LIST H — AEC BCSL657X	
BCSL657A	AUTOMATED SOFTWARE TESTING
BCSL657B	ETHICAL HACKING AND VULNERABILITY ASSESSMENT
BCSL657C	MLOPS
BCSL657D	SYSTEM DESIGN AND SCALABLE ARCHITECTURE
BCSL657E	COMPETITIVE PROGRAMMING
BCSL657F	MONITORING AND LOGGING SYSTEMS
BCSL657G	REINFORCEMENT LEARNING
BCSL657H	TIME SERIES ANALYSIS
BCSL657I	SERVERLESS COMPUTING
BCSL657J	SCALA AND JULIA



B.E. in Computer Science and Engineering : Autonomous Scheme and Syllabus

7th Semester

SCHEME - B - VII and VIII Semesters for the candidates who seek an internship with project work

SL NO	COURSE	COURSE CODE	COURSE TITLE	TD/PSB	TEACHING HOURS PER WEEK			(TW + SL) /Sem	EXAMINATION				CREDITS
					THEORY LECTURE (L)	TUTORIAL (T)	PRACTICAL (P)		DURATION IN HOURS	CIE MARKS	SEE MARKS	TOTAL MARKS	
1	IPCC	BCS701	TO BE COMPLETED IN 5TH/6TH SEMESTER	CSE	3	0	2	45	3	50	50	100	4
2	IPCC	BCS702	TO BE COMPLETED IN 5TH/6TH SEMESTER	CSE	3	0	2	45	3	50	50	100	4
3	PCC	BCS703	TO BE COMPLETED IN 5TH/6TH SEMESTER	CSE	3	2	0	45	3	50	50	100	4
4	PEC	BCS714X	PROFESSIONAL ELECTIVE COURSE (MOOC COURSES)	CSE	3	0	0	45	3	50	50	100	3
5	OEC	BXX755X	OPEN ELECTIVE COURSES (MOOC COURSES)	CSE AND ALLIED	3	0	0	45	3	50	50	100	3
1	PEC	BCS811X	PROFESSIONAL ELECTIVE COURSE (MOOC COURSES)	CSE	3	0	0	45	3	50	50	100	3
2	OEC	BXX852X	OPEN ELECTIVE COURSES (MOOC COURSES)	ANY	3	0	0	45	3	50	50	100	3
3	PROJ	BCS883	PROJECT-OUTCOME TRAINING	CSE	0	0	12	0	3	100	100	200	9
4	INT	BCS804	INTERNSHIP (INDUSTRY/RESEARCH - (02 SEMESTERS)	CSE	0	0	20	0	3	100	100	200	10
TOTAL										200	200	400	42



ESTD: 2001
An Institute with a Difference

RN Shetty Trust®

RNS INSTITUTE OF
TECHNOLOGY

An Autonomous Institution, Affiliated to VTU,
Accredited with NAAC A+ Grade

Department of Computer Science
and Engineering

Autonomous Scheme (Effective from Academic
Year 2024 – 25)



PROFESSIONAL ELECTIVE — BCS714X (MOOC COURSE)

BCS714X	BOS WILL PUBLISH COURSES BASED ON THE AVAILABILITY
---------	--

OPEN ELECTIVE — BXX755X (MOOC COURSE)

BXX755X	BOS WILL PUBLISH COURSES BASED ON THE AVAILABILITY
---------	--

PROFESSIONAL ELECTIVE — BCS811X (MOOC COURSE)

BCS811X	BOS WILL PUBLISH COURSES BASED ON THE AVAILABILITY
---------	--

OPEN ELECTIVE — BXX852X (MOOC COURSE)

BXX852X	BOS WILL PUBLISH COURSES BASED ON THE AVAILABILITY
---------	--

Contents

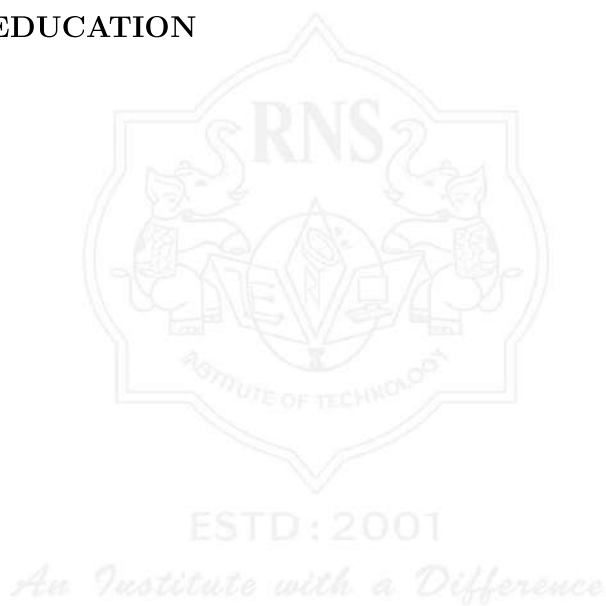
SEMESTER – V

1	SOFTWARE ENGINEERING AND PROJECT MANAGEMENT	5
2	COMPUTER NETWORKS	9
3	FULL STACK DEVELOPMENT	23
4	ADVANCED ALGORITHMS	28
5	BUSINESS INTELLIGENCE	31
6	UNIX SYSTEM PROGRAMMING	35
7	HADOOP AND SPARK	38
8	ETL AND DATA WAREHOUSING	42
9	EDGE AI AND TINY ML	46
10	DATA GOVERNANCE AND SECURITY	49
11	SITE RELIABILITY ENGINEERING	53
12	API DEVELOPMENT AND MICROSERVICES	57
13	INTRODUCTION TO COMPUTER FORENSICS	60
14	BIOLOGY FOR ENGINEERS	64
15	RESEARCH METHODOLOGY AND IPR	67
16	NATIONAL SERVICE SCHEME	71
17	YOGA	76

SEMESTER – VI

18 CLOUD COMPUTING	79
19 ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING	87
20 QUANTUM COMPUTING	91
21 MULTI-AGENT SYSTEMS	95
22 AUGMENTED REALITY AND VIRTUAL REALITY	99
23 CRYPTOGRAPHY AND NETWORK SECURITY	103
24 SECURE PROGRAMMING	107
25 DESIGN PATTERNS AND CLEAN CODE	111
26 APPLIED STATISTICAL MODELLING	114
27 PUBLIC CLOUD ARCHITECTURE	118
28 INTRODUCTION TO DATA CENTER NETWORKING	122
29 AI DRIVEN NETWORKS	126
30 DEVOPS AND CI/CD PIPELINES	130
31 INTRODUCTION TO DATA STRUCTURES	134
32 FUNDAMENTALS OF OPERATING SYSTEMS	138
33 INTRODUCTION TO DBMS	142
34 FOUNDATIONS OF DATA SCIENCE	145
35 ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING LAB- ORATORY	148
36 AUTOMATED SOFTWARE TESTING	153
37 ETHICAL HACKING AND VULNERABILITY ASSESSMENT	158
38 MLOPS	162
39 SYSTEM DESIGN AND SCALABLE ARCHITECTURE	167
40 COMPETITIVE PROGRAMMING	171
41 MONITORING AND LOGGING SYSTEMS	174

42 REINFORCEMENT LEARNING	179
43 TIME SERIES ANALYSIS	183
44 SERVERLESS COMPUTING	187
45 SCALA AND JULIA	190
46 INDIAN KNOWLEDGE SYSTEMS	194
47 NATIONAL SERVICE SCHEME	196
48 PHYSICAL EDUCATION	198
49 YOGA	201



SEMESTER – VII

50 HIGH PERFORMANCE COMPUTING	204
51 DATA ENGINEERING	212
52 BLOCKCHAIN TECHNOLOGY	220
53 EXPLAINABLE SOFT COMPUTING	224
54 DEEP LEARNING	228
55 COMPUTER VISION AND IMAGE PROCESSING	231
56 NATURAL LANGUAGE PROCESSING AND GEN AI	235
57 CLOUD SECURITY AND ZERO TRUST ARCHITECTURE	239
58 NETWORK DESIGN FUNDAMENTALS	243
59 UI/UX DESIGN FUNDAMENTALS	247
60 CLOUD PROFESSIONAL MACHINE LEARNING ENGINEER	250
61 INTRODUCTION TO ALGORITHMS	254
62 FOUNDATIONS OF INFORMATION SECURITY	258
63 FULL STACK WEB DEVELOPMENT	261
64 INTRODUCTION TO MACHINE LEARNING	265

V SEMESTER SUBJECTS

ESTD: 2001
An Institute with a Difference

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	SOFTWARE ENGINEERING AND PROJECT MANAGEMENT		
Course Code	BCS501	CIE MARKS	50
(L : T : P) + SL	(3:2:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	4	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Outline software engineering principles and activities involved in building large software programs. Identify ethical and professional issues and explain why they are of concern to Software Engineers.
2. Describe the process of requirement gathering, requirement classification, requirement specification, requirements validation and agile techniques.
3. Recognize the importance of Project Management with its methods and methodologies.
4. Identify software quality parameters and quantify software using measurements and metrics. List software quality standards and outline the practices involved.

Module-1	9 Hours
Software and Software Engineering: The nature of Software, Defining the Discipline, The software Process, Software Engineering Practice. Process Models: A generic process model, Defining a Framework Activity, Identifying a Task Set. Prescriptive process models Waterfall model, Prototyping process models, Evolutionary process model, Unified Process model, Process and Product. Textbook 1: Chapter 1 (1.1 - 1.4), Chapter 2 (2.1 - 2.3, 2.5 - 2.6)	
Module-2	9 Hours
Understanding Requirements: Requirements Engineering, Establishing the ground-work, Requirements Gathering, Developing use cases, Building Analysis Model, Negotiating Requirements, Requirements Monitoring, Validating Requirements. Requirements Modeling - A Recommended Approach: Requirement Analysis, Scenario-based modelling, Class-Based Modeling, Functional Modeling, Behavioural Modeling. Textbook 1: Chapter 7 (7.1 - 7.5, 7.7, 7.8), Chapter 8 (8.1 - 8.5)	

Module-3	9 Hours
Agile and Process: What is Agility? Agility and the cost of change. What is an Agile Process? Scrum, Other Agile Framework. Principles that guide practice: Core Principles and Framework Activities. Mobility and Agile Change Management: e-Change control, Content Management, Integration and Publishing, Version Control, Auditing and Reporting. Textbook 1: Chapter 3 (3.1 - 3.5), Chapter 6 (6.1 - 6.2), Chapter 22 (22.6)	
Module-4	9 Hours
Introduction to Project Management: Introduction, What is a project? Project and Importance of Project Management, Software Project versus other types of projects, Contract Management, Activities Covered by Software Project Management, Plans, Methods and Methodologies, Some ways of categorizing Software Projects, Stakeholders, Setting Objectives, Business Case, Project Success and Failure, Management and Management Control, Traditional versus Modern Project Management Practices. Project Evaluation and Program Management: Evaluation of Individual projects, Cost-benefit Evaluation Techniques, Risk Evaluation. Textbook 2: Chapter 1 (1.1 - 1.15), Chapter 2 (2.4 - 2.6)	
Module-5	9 Hours
Software Quality: Introduction, The place of software quality in project planning, Importance of software quality, Defining software quality, product versus process quality management, Process Capability Models, Techniques to Help Enhance Software Quality, Testing, Software Reliability, Quality plans. Software Effort Estimation: Software Effort Estimation Techniques, Bottom-up Estimation, Top-down approach, Expert Judgement, Albrecht Function Point Analysis, COSMIC Full function points, COCOMO II: Parametric Productivity Model, Cost Estimation, staffing Pattern. Textbook 2: Chapter 13 (13.1 - 13.4, 13.7, 13.9 - 13.13), Chapter 5 (5.5 - 5.8, 5.10, 5.12 - 5.15)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Understand the fundamental concepts of software, software engineering, software characteristics and various software process models used in software development.
CO2	Apply requirements engineering principles to elicit, analyse, specify, validate and manage software requirements.
CO3	Apply agile development principles, methodologies and frameworks to manage software development processes.

CO4	Analyse software project management concepts and stakeholder management for successful project execution.
CO5	Analyse software quality attributes, software quality management practices and software effort estimation techniques to evaluate software quality and project feasibility.

ASSESSMENT AND EVALUATION REGULATIONS OF HSMS

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Roger S. Pressman & Bruce R. Maxim, Software Engineering: A Practitioner's Approach, 9th Edition, McGraw Hill 2023 July.
2. Bob Hughes, Mike Cotterell, Rajib Mall, Software Project Management, 5th Edition 2009.

Reference Books:

1. Pankaj Jalote, An Integrated Approach to Software Engineering, Wiley India 2005.

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	COMPUTER NETWORKS		
Course Code	BCS502	CIE MARKS	50
(L : T : P) + SL	(3:0:2)+(45 Hours/Sem)	SEE MARKS	50
Credits	4	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Study the TCP/IP protocol suite, Switching criteria and Medium Access Control protocols for reliable and noisy channels.
2. Learn network layer services and IP versions.
3. Discuss transport layer services and understand UDP and TCP protocols.
4. Demonstrate the working of different concepts of Application layer and multimedia networks.

Module-1	9 Hours
Introduction: Data Communications, Networks, Network Types. Network Models: Protocol Layering, TCP/IP Protocol suite, The OSI model. Introduction to Physical Layer: Transmission media: Guided Media, Unguided Media. Switching: Introduction, Circuit switched networks, Packet Switching. Textbook 1: Chapter 1 (1.1 - 1.3), Chapter 2 (2.1 - 2.3), Chapter 7 (7.2, 7.3), Chapter 8 (8.1 - 8.3)	
Module-2	9 Hours
Data Link Layer - Error Detection and Correction: Introduction, Block Coding, Cyclic Codes, Checksum. Data link control: DLC Services, Data link layer protocols, High Level Data Link Control. Media Access Control: Random Access, Controlled Access Wired LANs: Ethernet: Standard Ethernet, IEEE 802.3 Wireless LANs: Introduction, IEEE 802.11 Project. Textbook 1: Chapter 10 (10.1 - 10.4), Chapter 11 (11.1 - 11.3), Chapter 12 (12.1 - 12.2), Chapter 13 (13.2), Chapter 15 (15.1 - 15.2.2)	

Module-3	9 Hours
The Network layer - Data Plane: What's Inside a Router? Input Processing, Switching, Output Processing, Where Does Queuing Occur? The Internet Protocol: IPv4 Addressing, IPv6 and More. The Network layer: Control Plane: Introduction, Routing Algorithms, Intra-AS routing in the Internet: OSPF, Routing Among the ISPs: BGP. Textbook 2: Chapter 4 (4.2, 4.3), Chapter 5 (5.1 - 5.4)	
Module-4	9 Hours
Transport Layer: Introduction and Transport-Layer Services, Multiplexing and Demultiplexing, Principles of Reliable Data Transfer, Connection-Oriented Transport TCP, Principles of Congestion Control, TCP Congestion Control. Textbook 2: Chapter 3 (3.1 - 3.7)	
Module-5	9 Hours
Application Layer: Principles of Network Applications, The Web and HTTP, Electronic Mail in the Internet, DNS, Peer-to-Peer file Distribution, Video Streaming and Content Distribution Networks, Socket Programming : Creating Network Applications. Textbook 2: Chapter 2 (2.1 - 2.7)	

PRACTICAL COMPONENT OF IPCC

PART A
Program 1: Error Detection in Data Link Layer using CRC-CCITT
Scenario: A logistics company uses a computer network to transmit shipment information between its regional warehouses and the central server. During transmission, data packets may be corrupted due to noise and interference on the communication channel. To ensure data integrity, the company decides to implement the CRC-CCITT (16-bit) error detection technique before accepting any received packet.
Program 2: Sliding Window Protocol for Reliable Data Transmission
Scenario: A multimedia streaming company provides live online lectures to thousands of students across different locations. To ensure smooth and reliable delivery of video data, the company uses a Sliding Window Protocol at the Data Link Layer. This protocol allows multiple frames to be sent before receiving acknowledgments, thereby improving channel utilization and transmission efficiency.

Program 3: Detecting Negative-Weight Cycles in a Regional Routing Network

Scenario:

A telecommunications company is designing a routing system for a regional network consisting of multiple routers connected through communication links. Each link between routers has an associated cost representing factors such as delay, bandwidth usage, or transmission expense. Due to changing network conditions, some links may have negative costs (representing incentives or optimized routes), but the network should be checked for routing loops caused by negative-weight cycles.

Program 4: TCP/IP Client-Server File Retrieval System

Scenario:

A university maintains a centralized document server that stores laboratory manuals, assignment files and course materials. Students access these resources through a client application running on their computers. When a student needs a specific file, the client sends the file name to the server over a TCP/IP network connection. The server checks whether the requested file exists in its repository. If the file is found, the server sends the contents of the file back to the client. Otherwise, it sends an appropriate error message indicating that the file is unavailable.

Program 5: Socket-Based client-server communication using UDP

Scenario:

A regional weather monitoring center broadcasts important weather alerts to remote observation stations located across different districts. Since the alerts are short text messages and need to be delivered quickly, the system uses a connectionless communication model based on UDP datagram sockets. An operator at the central server types weather updates and the connected monitoring stations receive and display the messages immediately.

Program 6: Leaky Bucket Algorithm for Network Traffic Congestion Control

Scenario:

A multimedia streaming company provides live online lectures to thousands of students simultaneously. During peak hours, the server receives bursts of data packets generated by video streams, user interactions and real-time notifications. If these packets are transmitted directly onto the network without regulation, congestion may occur, resulting in packet loss, increased delays and poor Quality of Service (QoS). To ensure smooth network operation, the company decides to implement a Leaky Bucket Algorithm, where incoming packets are temporarily stored in a finite-capacity bucket (buffer) and transmitted at a constant rate. Any packets arriving when the bucket is full are discarded.

PART B

Program 7: Network Configuration Testing and Routing Algorithms in Packet Tracer

Scenario: Test and verify Network configurations using Packet Tracer and explore static and dynamic routing algorithms.
Program 8: DHCP Service Configuration in a Simulated Network Topology
Scenario: Design a simple network topology and configure DHCP services using Cisco Packet Tracer.
Program 9: LAN Simulation and FTP Server Configuration
Scenario: Design and simulate a Local Area Network (LAN) using Cisco Packet Tracer and configure an FTP server to allow file transfer between the server and client systems.
Program 10: TCP/UDP Packet Sniffing Using Wireshark
Scenario: A network administrator is troubleshooting communication issues between a client and a server. To analyze the network traffic, the administrator uses Wireshark to capture both TCP and UDP packets. Capture the packets, apply appropriate filters to view TCP and UDP traffic, identify the source and destination IP addresses and port numbers and compare the communication characteristics of both protocols.
Open Ended Experiments
Program 11: HTTP Packet Sniffing Using Wireshark
Scenario: A web administrator wants to analyze the communication between a web browser and a web server to understand how HTTP requests and responses are exchanged. Using Wireshark, capture HTTP packets, apply an HTTP display filter, identify the request and response messages, source and destination IP addresses, port numbers and analyze the HTTP communication.
Program 12: NAT Configuration for Private-to-Public IP Translation
Scenario: Design and simulate a network using Cisco Packet Tracer and configure NAT services on a router to translate private IP addresses into public IP addresses.
Program 13: Inter-VLAN Routing Configuration
Scenario: Configure Inter VLAN network using Packet Tracer.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the fundamental concepts of data communication, network architectures, protocol layering, switching, and transmission technologies.
CO2	Demonstrate the use of error control, data link protocols, media access mechanisms, and LAN technologies for network communication.
CO3	Implement addressing, routing, transport-layer, and congestion-control mechanisms in computer network communication.
CO4	Examine the interactions among network, transport, and application-layer protocols for different communication scenarios.
CO5	Differentiate network protocols and technologies based on their characteristics, performance, and suitability for various applications.

ASSESSMENT AND EVALUATION REGULATIONS OF IPCC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE- IA Tests (A)	20	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 20 marks.
Continuous and Comprehensive Evaluation (B)	10	-	Any one Assessment methods as per 24RNBE4.2 scaled to 10 marks.
Total CIE Theory (A+B)	30	12	Marks of Tests and Assignments awarded for a maximum of 30.
CIE Practical	12	-	Marks awarded for Weekly Conduction of Experiments and Submission of Laboratory records, scaled to 12 marks.
CIE Practical Test	8	-	Average of marks of Two tests, each conducted for 100 marks covering all experiments, scaled to 8.
Total CIE Practical (C)	20	8	Marks of Experiments, Record and Test awarded for a maximum of 20 marks.
CIE: Theory + Practical (A + B + C)	50	20	CIE Evaluation is done for a maximum of 50 marks.

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
SEE (D)	50	18	SEE is a theory exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall be 3 hours.
CIE+SEE (A+B+C+D)	100	40	Final marks shall be the sum of marks scored in Components A, B, C and D.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE. The laboratory component of the IPCC shall be for CIE only. However, in SEE, the questions from the laboratory component shall be included in their respective modules only.			

Textbooks:

1. Behrouz A. Forouzan, Data Communications and Networking, 5th Edition, Tata McGraw-Hill, 2013.
2. James F. Kurose and Keith W. Ross, Computer Networking: A Top-Down Approach, 8th Edition, Pearson, 2023.

Reference Books:

1. Larry L. Peterson and Bruce S. Davie, Computer Networks – A Systems Approach, 4th Edition, Elsevier, 2019.
2. Nader F. Mir, Computer and Communication Networks, 2nd Edition, Pearson Education, 2015.
3. William Stallings, Data and Computer Communications, 10th Edition, Pearson Education, 2014.
4. Hans W. Barz and Gregory A. Bassett, Multimedia Networks: Protocols, Design and Application, 1st edition, Wiley, 2016.

Web links:

1. https://onlinecourses.nptel.ac.in/noc25_cs125/preview
2. <https://www.coursera.org/learn/introduction-to-networking-nvidia>

Self Study Topics:

1. Analyze various error detection and correction techniques.
2. Compare Distance Vector and Link State approaches, including Bellman-Ford and OSPF, with respect to convergence, scalability and routing efficiency.

3. Study performance characteristics of TCP and UDP protocols and examine their suitability for applications such as real-time streaming and reliable data transfer.

Activity Based Learning:

1. Certifications (Skill based): Industry-recognized certifications (NPTEL, CISCO, Infosys Springboard).



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	THEORY OF COMPUTATION AND COMPILER DESIGN		
Course Code	BCS503	CIE MARKS	50
(L : T : P) + SL	(3:0:2)+(45 Hours/Sem)	SEE MARKS	50
Credits	4	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand the fundamental concepts of Automata Theory, Formal Languages, Regular Expressions and the phases of Compiler Design.
2. Apply Finite Automata, Regular Expressions and Lexical Analysis techniques for recognizing regular languages and developing lexical analyzers.
3. Design Context-Free Grammars and Parsers for Syntax Analysis in Compiler Construction.
4. Analyze Pushdown Automata and Parsing Techniques used for Syntax Analysis in Compiler Construction.
5. Evaluate Computational Models, Syntax-Directed Translation Schemes and Code Generation Techniques in Compiler Implementation.

Module-1	9 Hours
Automata - The Methods and the Madness: The Central Concepts of Automata Theory. Finite Automata: Deterministic Finite Automata (DFA), Non-Deterministic Finite Automata (NFA), Finite Automata with Epsilon-Transitions. Properties of Regular Languages: Equivalence and Minimization of Automata. Introduction to Compiler Design: Language Processors, The Structure of a Compiler. Textbook 1: Chapter 1 (1.5), Chapter 2 (2.2 - 2.3, 2.5), Chapter 4 (4.4) Textbook 2: Chapter 1 (1.1 - 1.2)	
Module-2	9 Hours
Regular Expressions and Languages: Regular Expressions, Finite Automata and Regular Expressions. Properties of Regular Languages: Proving Languages Not to Be Regular. Lexical Analysis Phase of Compiler Design: The Role of the Lexical Analyzer, Input Buffering, Specification of Tokens, Recognition of Tokens. Textbook 1: Chapter 3 (3.1 - 3.2), Chapter 4 (4.1) Textbook 2: Chapter 3 (3.1 - 3.4)	

Module-3	9 Hours
Context Free Grammars and Languages: Context Free Grammars, Parse Trees, Ambiguity in Grammars and Languages. Syntax Analysis Phase of Compilers - Part-1: Introduction, Writing a Grammar: Eliminating Ambiguity, Elimination of Left Recursion, Left Factoring, Top-Down Parsing. Textbook 1: Chapter 5 (5.1.1 - 5.1.6), (5.2.1 - 5.2.2), (5.4) Textbook 2: Chapter 4 (4.1, 4.3 (4.3.2 - 4.3.4), 4.4)	
Module-4	9 Hours
Push Down Automata: Definition of the Pushdown Automaton, The Languages of a PDA. Syntax Analysis Phase of Compilers - Part-2: Bottom-up Parsing, Introduction to LR Parsing: Simple LR. Textbook 1: Chapter 6 (6.1 - 6.2) Textbook 2: Chapter 4 (4.5 - 4.6)	
Module-5	9 Hours
Introduction to Turing Machine: Problems that Computers Cannot Solve, The Turing Machine, Programming Techniques for Turing Machine, Extensions to the Basic Turing Machine. Other Phases of Compilers: Syntax Directed Translation: Syntax-Directed Definitions, Evaluation Orders for SDD's. Intermediate-Code Generation: Variants of Syntax Trees, Three-Address Code. Code Generation: Issues in the Design of a Code Generator. Textbook 1: Chapter 8 (8.1, 8.2, 8.3, 8.4) Textbook 2: Chapter 5 (5.1, 5.2), Chapter 6 (6.1, 6.2), Chapter 8 (8.1)	

PRACTICAL COMPONENT OF IPCC

(Automata + Compiler Structure)
Program 1: HR Access Control System (DFA) Scenario: An IT company wants to automatically validate employee login IDs before granting system access. Question: Design and implement a DFA to validate employee IDs that start with a capital letter, followed by 3–5 digits and end with _HR or _ENG. Display whether the ID is valid or invalid. Application: Used in enterprise authentication systems and structured credential validation.

Program 2: Smart Factory Command Handler (ϵ -NFA to DFA)

Scenario: A smart manufacturing unit allows operators to optionally issue an emergency override (EMG) before sending operational commands like START or STOP.

Question: Construct an ϵ -NFA for this command structure. Remove ϵ -transitions, convert it into an equivalent DFA and simulate the automaton for sample commands.

Application: Used in industrial control systems for flexible command validation.

Program 3: Highway Toll Controller Optimization (DFA Minimization)

Scenario: A digital toll gate controller is implemented using a DFA, but the hardware cost is high due to unnecessary states.

Question: Given the transition table of the toll gate DFA, minimize the automaton and produce the equivalent reduced DFA.

Application: Used in digital circuit optimization and embedded controller design.

Program 4: Visual Compiler Flow Demonstrator

Scenario: First-year compiler students struggle to understand how source code transforms into machine code.

Question: Develop a simulator that demonstrates how a source program passes through Lexical Analysis, Syntax Analysis, Semantic Analysis and Code Generation phases.

Application: Used in compiler education tools and programming language visualization software.

(Regular Expressions + Lexical Analysis)

Program 5: University Email Verification System (RE to NFA)

Scenario: A university portal must validate only official email addresses ending with @college.edu.

Question: Write a Regular Expression for institutional email validation, convert it into an NFA and simulate it for multiple email samples.

Application: Used in web form validation and institutional portal authentication.

Program 6: Secure Password Policy Analyzer (Non-Regular Language)

Scenario: A security team proposes a password rule requiring equal numbers of a and b characters in sequence.

Question: Attempt to design a DFA for $L = \{a^n b^n \mid n \geq 1\}$. Demonstrate why such a language cannot be recognized by a finite automaton.

Application: Illustrates the limitations of finite memory systems in cybersecurity policy design.

Program 7: ATM Transaction Log Scanner (LEX Program)

Scenario: A banking server processes thousands of ATM transaction logs daily and must classify tokens before further processing.

Question: Using LEX, design a lexical analyzer to identify keywords, account numbers, transaction amounts and operators from ATM log entries.

Application: Used in banking and financial transaction processing systems.

Program 8: Large File Source Reader (Input Buffering)

Scenario: A compiler must process very large source code files efficiently without frequent disk access.

Question: Implement a two-buffer input scheme to demonstrate efficient reading and lookahead capability in lexical analysis.

Application: Used in real-world compiler optimization for efficient file handling.

(CFG + Parsing)

Program 9: Customer Support Chat Syntax Validator (CFG)

Scenario: A chatbot system only accepts predefined structured requests such as "HELLO I NEED REFUND".

Question: Define a CFG for valid chatbot requests and generate parse trees to verify whether given inputs follow the grammar.

Application: Used in rule-based conversational AI systems.

Program 10: Billing Expression Parser (Ambiguity Detection)

Scenario: An online billing engine miscalculates expressions due to ambiguous grammar rules.

Question: Show ambiguity in the grammar for $id + id * id$, construct multiple parse trees and modify the grammar to remove ambiguity.

Application: Ensures correct operator precedence in programming languages and calculators.

Program 11: Online Discount Calculator (Predictive Parsing – LL(1))

Scenario: An e-commerce platform parses discount formulas entered by administrators.

Question: Construct FIRST and FOLLOW sets, build an LL(1) parsing table and implement a predictive parser for arithmetic expressions.

Application: Used in top-down parsing modules of compilers.

Program 12: Assignment Statement Validator (Shift-Reduce Parsing)

Scenario: A simple programming language interpreter must validate assignment statements before execution.

Question: Implement a shift-reduce parser to validate statements like $id = id + id$ and display stack operations.

Application: Used in bottom-up parsing mechanisms of compilers.

(PDA + LR Parsing)

Program 13: Database Transaction Nesting Checker (PDA)

Scenario: A database system logs nested transactions using BEGIN and END markers.

Question: Design and simulate a Pushdown Automaton to validate properly nested transaction logs.

Application: Used in database management systems to verify nested transactions.

Program 14: Programming Language Syntax Analyzer (SLR Parser)

Scenario: A compiler for a mini-language must parse arithmetic expressions efficiently.

Question: Construct LR (0) items, build the SLR parsing table and simulate parsing for sample input strings.

Application: Used in compiler construction for efficient bottom-up parsing.

(Turing Machine + SDD + Intermediate Code)

Program 15

Scenario: An embedded systems compiler must generate intermediate code for arithmetic expressions and also validate structured patterns beyond regular and context-free capabilities.

Question: (a) Using Syntax-Directed Definitions, generate Three Address Code for expressions like $a = b + c * d$. (b) Design and simulate a Turing Machine for the language $\{0^n 1^n\}$ and demonstrate its step-by-step execution.

Application: (a) Used in intermediate code generation phase of compilers. (b) Used to understand theoretical limits of computation and decidability.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Illustrate the concepts of Automata Theory, Formal Languages, Regular Expressions and Compiler Phases.
CO2	Apply Finite Automata and Regular Expressions for regular language recognition and implement a Lexical Analyzer using LEX.
CO3	Design Context-Free Grammars and Top-Down Parsers for Syntax Analysis.
CO4	Analyze Pushdown Automata and develop Bottom-Up Parsers for Syntax Analysis in Compiler Construction.
CO5	Examine computational models and compiler implementation techniques including syntax-directed translation and code generation.

ASSESSMENT AND EVALUATION REGULATIONS OF IPCC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE- IA Tests (A)	20	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 20 marks.
Continuous and Comprehensive Evaluation (B)	10	-	Any one assessment method as per 24RNBE4.2 scaled to 10 marks.
Total CIE Theory (A+B)	30	12	Marks of tests and assignments awarded for a maximum of 30.
CIE Practical	12	-	Marks awarded for Weekly Conduction of Experiments and Submission of Laboratory records, scaled to 12 marks.
CIE Practical Test	8	-	Average of marks of Two tests, each conducted for 100 marks covering all experiments, scaled to 8.
Total CIE Practical (C)	20	8	Marks of Experiments, Record and Test awarded for a maximum of 20 marks.
CIE: Theory + Practical (A+B+C)	50	20	CIE Evaluation is done for a maximum of 50 marks.
SEE (D)	50	18	SEE is a theory exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall be 3 hours.
CIE+SEE (A+B+C+D)	100	40	Final marks shall be the sum of marks scored in Components A, B, C and D.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE. The laboratory component of the IPCC shall be for CIE only. However, in SEE, the questions from the laboratory component shall be included in their respective modules only.			

Textbooks:

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Introduction to Automata Theory, Languages and Computation, 3rd Edition, Pearson, 2006.

-
2. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Compilers: Principles, Techniques and Tools, 2nd Edition, Pearson, 2006.

Reference Books:

1. Elaine Rich, Automata, Computability and Complexity, 1st Edition, Pearson Education, 2018.
2. K. L. P. Mishra, N. Chandrashekar, Theory of Computer Science, 3rd Edition, PHI, 2012.
3. Peter Linz, An Introduction to Formal Languages and Automata, 3rd Edition, Narosa Publishers, 1998.
4. K. Muneeswaran, Compiler Design, Oxford University Press, 2013.

Web Links:

1. <https://www.jflap.org/>



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	FULL STACK DEVELOPMENT		
Course Code	BCSL504	CIE MARKS	50
(L : T : P)	(0:0:2)	SEE MARKS	50
Credits	1	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand core JavaScript concepts for modern web development.
2. Design responsive web interfaces using Bootstrap and CSS frameworks.
3. Develop dynamic front-end applications using React JS.
4. Implement scalable back-end services using Express JS and RESTful APIs.
5. Model, store and manage data using NoSQL databases with industry best practices.

Program 1 - JavaScript Basics AND Functions

- a. Write JavaScript programs to perform basic arithmetic operations and display results using alerts and the console.
- b. Create arrays and perform operations such as insertion, deletion, searching and sorting.

Scenario:

A supermarket is digitizing its billing process. The cashier enters item details and the system must automatically handle price calculations, apply discounts for bulk purchases and display the final payable amount in a user-friendly manner. Additionally, the store maintains a dynamic list of item prices that frequently changes as products are added or removed.

Program 2 - JavaScript Objects AND Control Structures

- a. Perform string operations such as length, substring, replace, split and reverse.
- b. Implement functions, including arrow functions and write programs to check for a palindrome and to compute factorial.

Scenario:

A customer feedback system receives text messages from users. These messages need to be processed to extract useful insights, filter unwanted words and analyze patterns in the text. Meanwhile, the system must also validate certain user inputs and perform calculations based on given conditions.

Program 3 - DOM Manipulation

Create a web page that dynamically modifies content, styles and attributes using DOM methods.

Scenario:

A college portal displays student details on a webpage. The administration wants the page to update dynamically when different actions are performed, such as changing displayed information, highlighting important data, or updating visual elements without reloading the page.

Program 4 - Event Handling AND Browser Storage

Implement event handling including button clicks, mouse events, keyboard events and demonstrate event bubbling and delegation.

Scenario:

An online learning platform requires interactive features where user actions such as clicks, typing and mouse movements trigger different responses. Some elements are added dynamically and the system must efficiently handle interactions for both existing and newly created elements while demonstrating how events propagate through the interface.

Program 5 - React Components AND Props

Develop a React application to display a static list (e.g., issues, products, or students) using functional components and props.

Scenario:

A school is developing a web interface to display structured information about its students. The system should be modular, reusable and capable of passing data efficiently between different parts of the application while maintaining clarity in the UI design.

Program 6 - React State AND Hooks

Build a React component using state and hooks to implement a counter, form input handling and conditional rendering.

Scenario:

A simple task management interface is required where users can interact with the system to update values, provide inputs and view changes instantly. The interface should respond differently based on the current data and reflect updates dynamically without refreshing the page.

Program 7 - API Integration with React

Develop a React application that fetches data from a public REST API and displays it using the `useEffect` hook.

Scenario:

A dashboard application needs to present real-time information fetched from an external service. The data should be retrieved when the application loads and displayed in a meaningful format, ensuring that the interface updates automatically based on the received data.

Program 8 - Backend Development using Express

Build a full-stack mini application using Express to create RESTful endpoints and connect them to a React frontend.

Scenario:

A startup is building the backend for its application where data must be handled through well-structured endpoints. The system should support adding new data, retrieving existing records and managing changes efficiently while ensuring smooth communication with the frontend.

Program 9 - MongoDB CRUD Operations

Use MongoDB and Mongoose to implement CRUD operations with schema validation and connect them to Express APIs.

Scenario:

An organization needs to store and manage structured records in a database. The system should ensure that data can be created, viewed, modified and deleted while maintaining consistency and validating the stored information according to predefined rules.

Program 10 - Full Stack Application Deployment (Advanced)

Deploy a MERN stack application with environment variables, authentication (JWT/session) and demonstrate cloud hosting with CI/CD basics.

Scenario:

A company plans to launch its web application for public use. The system must securely manage user access, protect sensitive information and be deployed in a way that allows continuous updates and reliable access over the internet.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Apply JavaScript to build dynamic and interactive web applications.
CO2	Design responsive front-end interfaces using modern frameworks.
CO3	Develop server-side applications using Express and RESTful APIs.
CO4	Apply NoSQL database concepts to design, store and manage data effectively.
CO5	Implement full-stack applications using industry practices.

ASSESSMENT AND EVALUATION REGULATIONS OF PCCL

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE Practical (A)	30	-	Marks awarded for Weekly Conduction of Experiments and Submission of Laboratory records, scaled to 30 marks.
CIE Practical Test (B)	20	-	Average of marks of Two tests, each conducted for 100 marks covering all experiments, scaled to 20
Total CIE Practical (A+B)	50	20	Sum of Marks of Experiments, Record and Test shall be CIE marks awarded for a maximum of 50.
SEE (C)	50	20	SEE is a lab exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE+SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Jon Duckett, JavaScript & jQuery: Interactive Front-End Web Development, 1st Edition, Wiley, 2014.
2. Vasan Subramanian, Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React and Node, 2nd Edition, Apress, 2019.

Reference Books:

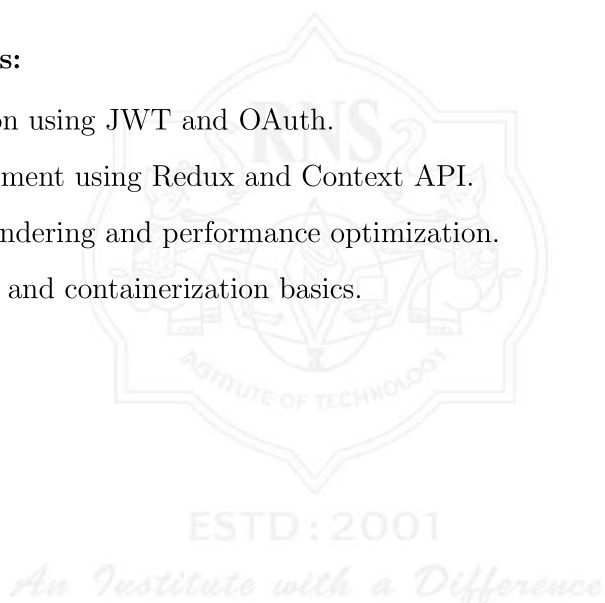
1. David Flanagan, JavaScript: The Definitive Guide, 7th Edition, O'Reilly Media, 2020.
2. Alex Banks and Eve Porcello, Learning React: Modern Patterns for Developing React Apps, 2nd Edition, O'Reilly Media, 2020.
3. Leonard Richardson and Mike Amundsen, RESTful Web APIs, 1st Edition, O'Reilly Media, 2013.

Web Links:

1. <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
2. <https://react.dev/learn>
3. <https://expressjs.com/>
4. <https://www.mongodb.com/docs/>
5. <https://nodejs.org/en/docs/>
6. <https://mongoosejs.com/docs/>

Self Study Topics:

1. Authentication using JWT and OAuth.
2. State management using Redux and Context API.
3. Server-side rendering and performance optimization.
4. Microservices and containerization basics.



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Chapterwise Based Credit System (CBCS)
SEMESTER – V

Course Title	ADVANCED ALGORITHMS		
Course Code	BCS515A	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Develop a strong foundation in amortized analysis and complexity evaluation of data structures.
2. Gain knowledge of string-matching techniques and prediction algorithms.
3. Enhance problem-solving skills through concepts such as flow networks, bipartite matching and Discrete Fourier Transform (DFT).
4. Build understanding of fundamental number-theoretic algorithms.
5. Strengthen algorithm design skills, including dynamic programming, randomized algorithms and approximation techniques.

Module-1	9 Hours
Growth of Functions: Asymptotic Notations, Standard notations and common functions. Divide and Conquer: The Substitution method for solving recurrences, Recurrence Tree method for solving recurrences, Master method for solving recurrences. Amortized Complexity Analysis: Aggregate, Accounting and Potential Methods. Textbook : Chapter 3 (3.1 - 3.2), Chapter 4 (4.3 - 4.5), Chapter 17 (17.1 - 17.3)	
Module-2	9 Hours
String Matching: Naïve String Matching, Boyer–Moore, Rabin–Karp, String matching with Finite State Automata and Knuth–Morris–Pratt Algorithms. Textbook : Chapter 32 (32.1 - 32.4)	
Module-3	9 Hours
Maximum Flow, Polynomials and FFT: Flow Networks, Ford-Fulkerson method, Edmonds-Karp algorithm, Maximum Bi-Partite Matching. Polynomials and FFT: Representation of Polynomials, Efficient Implementation of FFT, DFT and FFT. Textbook : Chapter 26 (26.1 - 26.3), Chapter 30 (30.1 - 30.3)	

Module-4	9 Hours
Number-Theoretic Algorithms: Elementary number-theoretic notions, Greatest Common Divisor, Modular Arithmetic, Solving modular linear equations, Modular Inverse, The Chinese remainder theorem, Powers of an element, The RSA public-key cryptosystem, Primality testing, Integer factorization. Textbook : Chapter 31 (31.1 - 31.9)	
Module-5	9 Hours
Dynamic Programming, Randomized Algorithms and Approximation Algorithms: Rod-Cutting, Matrix-Chain Multiplication, Elements of Dynamic Programming. Randomized Algorithms: Hiring Problem, Indicator random variables. Approximation Algorithm: Vertex Cover Problem, Randomization and linear programming. Textbook : Chapter 15 (15.1 - 15.3), Chapter 5 (5.1, 5.2), Chapter 35 (35.1 - 35.4)	

COURSE OUTCOMES:

After successfully completing the course, the student will be able to:

CO1	Analyse complex data structures using amortized analysis techniques.
CO2	Design and apply efficient string-matching algorithms to solve computational string-processing challenges.
CO3	Implement maximum flow and fast Fourier transform (FFT) algorithms.
CO4	Apply number-theoretic concepts to design cryptographic algorithms.
CO5	Analyse complex problems using dynamic programming, randomized algorithms and approximation algorithms.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.

Total CIE (A+B)	50	20	CIE evaluation is done for a maximum of 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. T H Cormen, C E Leiserson, R L Rivest and C Stein: Introduction to Algorithms, PHI, 3rd Edition, 2010.

Reference Books:

1. Steven Skiena: The Algorithm Manual , Springer, 2nd Edition, Springer, 2008.
2. Motwani and P Raghavan: Randomized Algorithms, Cambridge University Press, 2011.

Web links:

1. <https://ocw.mit.edu/courses/6-046j-design-and-analysis-of-algorithms>
2. <https://sl1nk.com/kta7wbb>

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	BUSINESS INTELLIGENCE		
Course Code	BCS515B	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The objectives of the course are to facilitate the learners to:

1. Understand the nature of data, statistical Modelling and visualization.
2. Learn concepts of Business Intelligence and Data Warehousing.
3. Gain knowledge on Data mining process and SNA, text and Web analytics
4. Analyze unstructured data using NLP and web analytics techniques to extract, interpret and derive actionable business insights.
5. Apply NLP and web analytics to analyse unstructured data for business insights.

Module-1

9 Hours

An Overview of Business Intelligence, Analytics and Decision Support: Changing Business Environments and Computerized Decision Support, Managerial Decision Making, Information system support for Decision Making, An early Framework for Computerized Decision Support, The concept of Decision Support System (DSS), A Framework for Business Intelligence.

Foundations and Technologies for Decision Making: Decision Making: Introduction and Definitions, Phases of Decision-making process, Decision making: Intelligence phase, Decision making: Design Phase, Decision making: Implementation Phase, Decision making: The Choice Phase, How Decisions are supported, Decision Support Systems: Capabilities.

Textbook: Chapter 1 (1.2 - 1.7), Chapter 2 (2.2 - 2.9)

Module-2

9 Hours

Descriptive Analytics - Data Warehousing: Data Warehousing Definitions and Concepts, Data Warehousing Process Overview, Data Warehousing Architectures, Data Integration and the Extraction, Transformation and Load (ETL) Processes, Data Warehouse Development, Data Warehousing Implementation Issues, Real-Time Data Warehousing, Data Warehouse Administration, Security Issues and Future Trends.

Business Reporting, Visual Analytics: Business Reporting Definitions and concepts, Data and information Visualization, Different types of charts and Graphs Visualization.

Textbook: Chapter 3 (3.2 - 3.9), Chapter 4 (4.2 - 4.4)

Module-3	9 Hours
Predictive Analytics - Data mining: Data mining Concepts and Applications, Data mining Applications, Data mining Process, Data mining Methods. Prescriptive Analytics - Optimization: Model-Based Decision-Making, Structure of Mathematical Models for Decision Support, Certainty, Uncertainty and Risk, Decision Modeling with Spreadsheets. Textbook: Chapter 5 (5.2 - 5.5), Chapter 9 (9.2 - 9.5)	
Module-4	9 Hours
Predictive Analytics – Text Analytics, Text Mining and Sentiment Analysis: Text Analytics and Text Mining Concepts and Definitions, Natural Language Processing (NLP), Text Mining Applications, Text Mining Process, Text Mining Tools, Sentiment analysis overview, Sentiment analysis Applications, Sentiment analysis Process. Web Analytics and Social Media Analytics: Web Mining Overview, Web Content and Web Structure Mining, Search Engines, Search Engine Optimization, Web Usage Mining (Web Analytics), Social Analytics and Social Network Analysis, Social Media Definitions and Concepts, Social Media Analytics. Textbook: Chapter 7 (7.2 - 7.9), Chapter 8 (8.2 - 8.6, 8.8 - 8.10)	
Module-5	9 Hours
Prescriptive Analytics - Heuristics Search Methods and Simulation: Problem Solving search Methods, Genetic Algorithms and GA Applications, Simulation, Visual Interactive Simulation, System Dynamics Modelling, Agent Based Modelling. Automated Decision Systems and Expert Systems: Automated Decision Systems, Artificial Intelligence Field, Basic concepts of Expert Systems, Applications of Expert Systems, Structure of Expert Systems, Knowledge Engineering. Textbook: Chapter 10 (10.2 - 10.7), Chapter 11 (11.2 - 11.7)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the role of business intelligence and analytics in a dynamic business environment.
CO2	Apply modern tools for statistical modelling and visualization.
CO3	Demonstrate descriptive analytics for business intelligence and data warehousing.
CO4	Implement algorithms for data mining techniques and processes.
CO5	Implement scripts for text and web mining and social network analysis.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Ramesh Sharda, Dursun Delen and Efraim Turban, Business Intelligence, Analytics, Data Science and AI – A Managerial Perspective, 5th edition, Global Edition, Pearson Education Limited, 2024.

Reference Books:

1. Steve Williams, Business Intelligence Strategy and Big Data Analytics - A General Management Perspective, 1st Edition, Morgan Kaufmann (Elsevier) , 2016.
2. Vincent Charles, Pratibha Garg, Neha Gupta and Mohini Agarwal, Data Analytics and Business Intelligence - Computational Frameworks, Practices and Applications, 1st Edition, CRC Press 2023.
3. Ira J. Haimowitz, Data Analytics for Business - Lessons for Sales, Marketing and Strategy, 1st Edition , Routledge (Taylor and Francis), 2023.

Self Learning Activities:

1. Comparative Study of BI Architectures and Data Warehousing Models: Investigating the architectural differences between centralized data warehouses, data marts and modern cloud-based lakehouse architectures, analyzing trade-offs in scalability, performance, real-time processing and cost efficiency.
2. Hands-on BI Tool Implementation and Dashboard Development: Developing practical expertise in tools such as Tableau, Power BI, or Google Data Studio to design interactive dashboards, implement KPIs and perform drill-down analytics using real-world datasets.
3. ETL Process Design and Data Integration Strategies: Exploring the end-to-end ETL (Extract, Transform, Load) pipeline by integrating heterogeneous data sources, performing data cleansing, transformation and loading into a structured warehouse environment.
4. Predictive and Prescriptive Analytics using Data Mining Techniques: Implementing classification, clustering and regression algorithms using Python (Scikit-learn) or RapidMiner to generate predictive insights and applying optimization models for prescriptive decision-making.
5. Text, Web and Social Media Analytics for Business Insights: Analyzing unstructured data through sentiment analysis, topic modeling and web usage mining using NLP libraries such as NLTK or spaCy to derive customer behavior patterns and strategic insights.

Web links:

1. <https://www.tableau.com/business-intelligence/what-is-business-intelligence>
2. https://onlinecourses.nptel.ac.in/noc24_cs65/preview
3. https://onlinecourses.nptel.ac.in/noc21_cs45/preview
4. <https://cloud.google.com/learn/what-is-business-intelligence>
5. <https://www.geeksforgeeks.org/what-is-data-analytic>

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	UNIX SYSTEM PROGRAMMING		
Course Code	BCS515C	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand UNIX standards and UNIX/POSIX programming interfaces.
2. Learn how UNIX files and file systems work.
3. Analyse UNIX process creation and process control mechanisms.
4. Demonstrate signal handling and daemon process programming techniques.
5. Compare and evaluate various inter process communication techniques in UNIX.

Module-1	9 Hours
Introduction: UNIX and ANSI Standards: The ANSI C Standard, The ANSI/ISO C++ Standards, Difference between ANSI C and C++, The POSIX Standards, The POSIX.1 FIPS Standard, The X/Open Standards. UNIX and POSIX APIs: The POSIX APIs, The UNIX and POSIX Development Environment, API Common Characteristics. Textbook 1: Chapter 1, Chapter 5	
Module-2	9 Hours
UNIX Files and APIs: File Types, The UNIX and POSIX File System, The UNIX and POSIX File Attributes, Inodes in UNIX System V, Application Program Interface to Files, UNIX Kernel Support for Files, Relationship of C Stream Pointers and File Descriptors, Directory Files, Hard and Symbolic Links. UNIX File APIs: General File APIs, File and Record Locking, Directory File APIs, Device File APIs, FIFO File APIs, Symbolic Link File APIs. Textbook 1: Chapter 6, Chapter 7 (7.1 - 7.6)	
Module-3	9 Hours
Process Environment: Introduction, main function, Process Termination, Command-Line Arguments, Environment List, Memory Layout of a C Program, Shared Libraries, Memory Allocation, Environment Variables, setjmp and longjmp Functions, getrlimit, setrlimit Functions.	

Process Control: Introduction, Process Identifiers, fork Function, vfork Function, exit Functions, wait and waitpid Function, wait3 and wait4 Functions, Race Conditions, exec Functions.

Textbook 2: Chapter 7, Chapter 8 (8.1 – 8.6, 8.8 - 8.10)

Module-4

9 Hours

Signals and Daemon Processes: The UNIX Kernel Support for Signals, signal, Signal Mask, sigaction, The SIGCHLD Signal and the waitpid Function, The sigsetjmp and siglongjmp APIs, Kill, Alarm, Interval Timers, POSIX.1b Timers.

Daemon Processes: Introduction, Daemon Characteristics, Coding Rules, Error Logging, Client-Server Model.

Textbook 1: Chapter 9

Textbook 2: Chapter 13 (13.1 - 13.4, 13.7)

Module-5

9 Hours

Interprocess Communication: Introduction: Pipes, popen, pclose Functions, Co-processes, FIFOs, Message Queues, Semaphores, Shared Memory, Client-Server Properties.

Textbook 2: Chapter 15 (15.2 - 15.5, 15.7 - 15.10)

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Understand UNIX, ANSI, POSIX and X/Open standards and identify appropriate UNIX/POSIX APIs.
CO2	Apply UNIX file system concepts and APIs to create and manage files, directories and links.
CO3	Analyse UNIX process creation, execution, termination and process control mechanisms.
CO4	Demonstrate signal handling and daemon process implementation in UNIX systems.
CO5	Implement interprocess communication using pipes, FIFOs, shared memory and message queues.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of two Internal Assessment Tests shall be scaled down to 30 marks.

Continuous and Comprehensive Evaluation (B)	20	-	Any two assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation is done for a maximum of 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A + B + C)	100	40	Final marks shall be the sum of marks scored in components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Unix System Programming Using C++ - Terrence Chan, PHI, 1st Edition, 1997.
2. Advanced Programming in the UNIX Environment - W. Richard Stevens, Stephen A. Rago, 3rd Edition, Pearson Education PHI, 2005.

Reference Books:

1. Advanced Unix Programming - Marc J. Rochkind, 2nd Edition, Pearson Education, 2005.
2. The Design of the UNIX Operating System - Maurice J. Bach, Pearson Education PHI, 1987.
3. Unix Internals - Uresh Vahalia, Pearson Education, 2001.

Web links:

1. https://www.man7.org/training/lusp/course_outline.html
2. <https://www.geeksforgeeks.org/linux-unix/linux-system-call-in-detail/>
3. <https://codelucky.com/unix-system-calls-posix/>
4. <https://youtube.com/watch?v=hLGwgbrlIXM>
5. <https://youtube.com/watch?v=2DrjQBL5FMU>
6. <https://youtu.be/k4Ww6gFeF0M>

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	HADOOP AND SPARK		
Course Code	BCS515D	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Introduce students to the fundamentals of Big Data and the need for distributed computing using Hadoop. Learn Hadoop Ecosystem & Map Reduce
2. Explain Hadoop architecture, Map Reduce programming model and its role in large-scale data processing.
3. Understand HDFS & Data Handling Provide knowledge of Hadoop Distributed File System (HDFS), data storage, data flow and data integrity concepts.
4. Explore Data Processing Tools (Pig & Spark) Familiarize students with high-level data processing tools like Pig and Apache Spark for efficient data analysis.
5. Develop Skills in Distributed Data Processing Enable students to design and implement applications using Hadoop and Spark for real-world data problems.

Module-1	9 Hours
Meet Hadoop: Data, Data Storage and Analysis, Comparison with Other Systems, RDBMS, Grid Computing, Volunteer Computing, A Brief History of Hadoop, The Apache Hadoop Project. MapReduce: A Weather Dataset, Data Format, Analyzing the Data with Unix Tools, Analyzing the Data with Hadoop, Map and Reduce, Java MapReduce, Scaling Out, Data Flow, Combiner Functions, Running a Distributed Map Reduce Job, Hadoop Streaming, Ruby, Python, Hadoop Pipes Textbook 1: Chapter 1 (1.1 - 1.5), Chapter 2 (2.1 - 2.6)	
Module-2	9 Hours
The Hadoop Distributed Filesystem: The Design of HDFS, HDFS Concepts, Blocks, Name nodes and Data nodes, Basic Filesystem Operations, Hadoop Filesystems, Reading Data from a Hadoop URL, Data Flow, Anatomy of a File Read and Anatomy of a File Write Hadoop I/O: Data Integrity, Data Integrity in HDFS, Compression, Codecs, Compression and Input Splits, Using Compression in MapReduce Textbook 1: Chapter 3 (3.1 - 3.6), Chapter 4 (4.1 - 4.2)	

Module-3	9 Hours
Pig: Comparison with Databases, Pig Latin, Structure, Statements Expressions, Types, Schemas, Functions, User-Defined Functions, A Filter UDF, An Eval UDF, A Load UDF, Data Processing Operators, Loading and Storing Data, Filtering Data Grouping and Joining Data, Sorting Data, Combining and Splitting Data Textbook 1: Chapter 11 (11.1 - 11.6)	
Module-4	9 Hours
Introduction to Data Analysis with Spark: Introduction to Data Analysis with Spark., What Is Apache Spark? A Unified Stack Spark Core, Spark SQL, Spark Streaming, MLlib, GraphX, Cluster Managers Downloading Spark and Getting Started: Downloading Spark Python and Scala Shell, Introduction to Core Spark Concepts Standalone Applications, initializing a Spark Context, Building Standalone Applications Textbook 2: Chapter 1 (1.1 - 1.2), Chapter 2 (2.1 - 2.4)	
Module-5	9 Hours
Working with Key/Value Pairs: Motivation, Creating Pair RDDs, Transformations on Pair RDDs, Aggregations, Grouping Data, Joins, Sorting Data, Actions Available on Pair RDDs, Data Partitioning (Advanced), Determining an RDD's Partitioner, Operations That Benefit from Partitioning, Operations That Affect Partitioning , Example: PageRank Custom Partitioners, Conclusion Loading and Saving Your Data: Motivation, File Formats, Text Files, JSON, CommaSeparated Values and Tab-Separated Values, Sequence Files, Object Files Textbook 2: Chapter 4 (4.1 - 4.6), Chapter 5 (5.1 - 5.2)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain Hadoop Architecture, Hadoop Ecosystem components, MapReduce workflow and Hadoop's role in Big Data processing.
CO2	Apply HDFS concepts for file operations, data storage and data management.
CO3	Implement MapReduce programs using Java, Python, or Hadoop Streaming.
CO4	Apply Pig Latin commands and Spark RDD operations for large-scale data processing.
CO5	Analyze partitioning, key-value pairs and distributed processing techniques in Big Data applications.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE- IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted average of two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any one Assessment methods as per 24RNBE4.2 scaled to 10 marks.
Total CIE (A+B)	50	20	CIE Evaluation is done for a maximum of 50 marks.
SEE(C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be scaled be 3 hours.
CIE+SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in components A,B,C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Tom White, Hadoop, The Definitive Guide, 3rd Edition, O'Reilly Publications, 2012.
2. Matei Zaharia, Patrick Wendellandy Konwinski, Holden Karau, Learning Spark,1st Edition, O'Reilly Media,2015.

Reference Books:

1. Shashank Tiwari, Professional NoSQL, 1st Edition Wiley India Pvt. Ltd ,2011.
2. Kyle Banker ,Peter Bakkum, Shaun Verch, Douglas Garrett,Tim Hawkins,MongoDB in Action, DreamTech Press, 2nd Edition ,2016.
3. Chris Eaton,Paul Zikopoulos,Tom Deutsch,George Lapis,Dirk Deroos,Understanding Big Data : Analytics for Enterprise Class Hadoop and Streaming Data, 1st Edition, Mcgraw Hill Education (India)Pvt.Ltd.,2012.

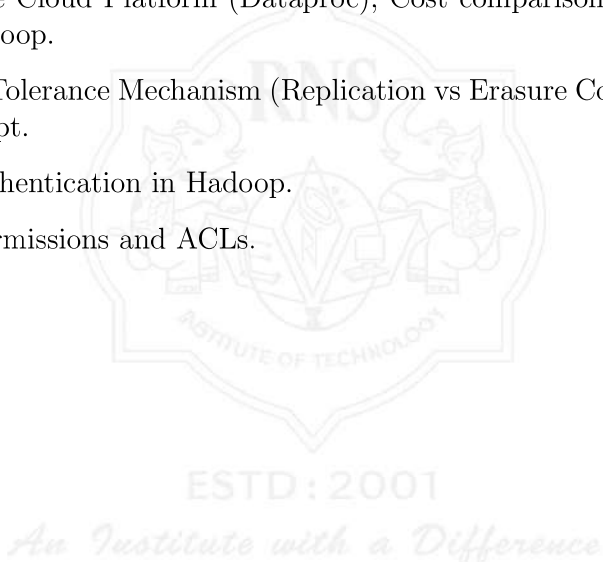
Web links:

1. <https://www.youtube.com/watch?v=HHR0-iJp2sM>
2. <https://0x0fff.com/hadoop-vs-mpp/>

3. <https://www.slideshare.net/yiranpang/random-notes-on-big-data-26439474>
4. <https://www.coursera.org/learn/big-data-introduction?specialization=big-data>
5. <http://www.bigdatauniversity.com>
6. <http://www.mongodb.com>
7. <http://hadoop.apache.org/>

Self Study Topics:

1. Hadoop on cloud platforms, Amazon Web Services (EMR), Microsoft Azure (HDInsight), Google Cloud Platform (Dataproc), Cost comparison: On-premise Hadoop vs Cloud Hadoop.
2. HDFS Fault Tolerance Mechanism (Replication vs Erasure Coding) and HDFS Federation concept.
3. Kerberos Authentication in Hadoop.
4. HDFS file permissions and ACLs.



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	ETL AND DATA WAREHOUSING		
Course Code	BCS515E	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The objectives of the course are to facilitate the learners to:

1. Understand the need, concepts and architecture of data warehousing and distinguish it from operational databases.
2. Apply design, business and technical considerations to build scalable and efficient data warehouses.
3. Analyze DBMS schemas, multidimensional models and ETL tools for effective decision support and data transformation.
4. Evaluate metadata management strategies and OLAP technologies for multidimensional analysis and performance optimization.
5. Create ETL workflows, staging structures and data quality processes to integrate heterogeneous sources and deliver reliable analytical data.

Module-1	9 Hours
Introduction to Data Warehousing: The Need for Data warehousing, Paradigm Shift, Business Problem Definition, Operational and Informational Data Stores, Data warehouse Definition and Characteristics, Data warehouse Architecture. Data Warehousing Components: Overall Architecture, Data Warehouse Database, Sourcing, Acquisition, Cleanup and Transformational Tools, Metadata, Access Tools, Data Marts, Data Warehouse Administration and Management. Textbook 1: Chapter 1 (1.2 - 1.7), Chapter 6 (6.1 - 6.7)	
Module-2	9 Hours
Building a Data Warehouse: Business Considerations, Design Considerations, Technical Considerations, Implementation Consideration, Integrated Solutions. Benefits of Data Warehousing. Mapping the Data Warehouse to a Multiprocessor Architecture: Relational Database Technology for Data Warehouse, Database Architecture for Parallel Processing, Parallel RDBMS features. Textbook 1: Chapter 7 (7.1 - 7.6), Chapter 8 (8.1 - 8.3)	

Module-3	9 Hours
DBMS Schema for Decision Support: Data Layout for Best Access, Multidimensional Data Model, Star Schema, STARjoin and STARindex, Bitmapped Indexing-SYBASE IQ, Column Local Storage. Data Extraction, Cleanup and Transformation Tools: Tool Requirements, Vendor Approaches, Access to Legacy Data, Vendor Solutions, Transformation Engines. Textbook 1: Chapter 9 (9.1 – 9.6), Chapter 10 (10.1 - 10.5)	
Module-4	9 Hours
Metadata: Metadata Defined, Metadata Interchange Initiative, Metadata Repository, Metadata Management, Implementation Examples, Metadata Trends. On-line Analytical Processing (OLAP): Need for OLAP, Multidimensional Data Model, OLAP Guidelines, Multidimensional versus Multirelational OLAP, Categorizing of OLAP Tools, State of the Market, OLAP Tools and the Internet. Textbook 1: Chapter 11 (11.1 - 11.6), Chapter 13 (13.1 - 13.5,13.7)	
Module-5	9 Hours
ETL Data Structures: To Stage or Not to Stage, Designing the Staging Area, Data Structures in the ETL System, Planning and Design Standards Extracting: The Logical Data Map, Inside the Logical Data Map, Building the Logical Data Map, Integrating Heterogeneous Data Sources Cleaning and Conforming: Defining Data Quality, Cleaning Deliverables Textbook 2: Chapter 2 (2.1 - 2.4), Chapter 3 (3.1 - 3.4), Chapter 4 (4.1, 4.4)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the fundamental concepts, components and architecture of data warehousing and its distinction from operational databases.
CO2	Apply ETL/ELT processes and OLAP operations in data warehousing environments.
CO3	Analyze metadata repositories, governance challenges, data marts and partitioning approaches in data warehouse systems.
CO4	Apply dimensional modeling techniques to design schemas and construct multidimensional data cubes.
CO5	Analyze OLAP models and partitioning strategies for query performance optimization.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE Evaluation is done for a maximum of 50 marks
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE+SEE(A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Alex Berson & Stephen J. Smith, Data Warehousing, Data Mining & OLAP, Tata McGraw-Hill Edition, 13th Reprint, 2008.
2. Ralph Kimball & Joe Caserta, The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleaning, Conforming and Delivering Data 1st Edition, 2004.

Reference Books:

1. Ralph Kimball, The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling, 3rd Edition, 2013.
2. Paul Raj Ponniah, Data Warehousing Fundamentals for IT Professionals, 2nd Edition 2012.
3. K.P. Soman, Shyam Diwakar and V. Ajay, Insight into Data Mining Theory and Practice, Eastern Economy Edition, 1st Edition, Prentice Hall of India, 2006.

Web Links:

1. <https://nptel.ac.in/courses/106106361>

Self Learning Activities

1. Comparative Analysis of Data Warehouse Architectures Investigate the trade-offs between traditional three-tier architectures, autonomous data warehouses and cloud-native platforms (e.g., Snowflake, Azure Synapse) in terms of scalability, performance and cost.
2. Hands-on ETL/ELT Pipeline Development Gain practical experience by building ETL workflows using tools like Talend, Informatica, or Apache NiFi and compare them with ELT approaches using SQL-based transformations in modern cloud warehouses.
3. Metadata Management Case Study Explore the role of metadata repositories in large-scale data warehouses, analyze categories of metadata and evaluate challenges in metadata governance and integration.
4. Schema Design and Modeling Exercise Design and simulate dimensional models (Star, Snowflake, Fact Constellation) using sample datasets and assess their impact on query performance and storage efficiency.
5. OLAP Operations Exploration Implement OLAP operations (slice, dice, roll-up, drill-down) on a sample cube using tools like Microsoft Analysis Services or open-source OLAP engines and compare ROLAP, MOLAP and HOLAP approaches.
6. Partitioning Strategy Experimentation Apply vertical, horizontal and row-splitting partition strategies on large datasets and measure their effect on query optimization and system performance.
7. System & Process Manager Simulation Configure a mock data warehouse environment to test system managers (scheduling, backup/recovery) and process managers (load, query, tuning), evaluating their role in reliability and efficiency.

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	EDGE AI AND TINY ML		
Course Code	BCS515F	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand the fundamentals of Edge AI and TinyML, including embedded systems, IoT, artificial intelligence and machine learning concepts.
2. Explain different types of sensors, signals and hardware architectures used in Edge AI systems.
3. Analyze real-world applications of Edge AI, including responsible design and ethical considerations.
4. Develop basic machine learning models using Python, TensorFlow and Jupyter/Google Colab for TinyML applications.
5. Implement and deploy TinyML models on embedded systems, understanding model conversion and application development.

Module-1	9 Hours
A Brief Introduction to Edge AI: Defining Key Terms, Embedded ,The Edge (and the Internet of Things) ,Artificial Intelligence ,Machine Learning ,Edge AI, Embedded Machine Learning and Tiny Machine Learning, Digital Signal Processing, Why Do We Need Edge AI?, To Understand the Benefits of Edge AI, Just BLERP, Edge AI for Good, Key Differences Between Edge AI and Regular AI. Textbook 1: Chapter 1	
Module-2	9 Hours
The Hardware of Edge AI: Sensors, Signals and Sources of Data ,Types of Sensors and Signals ,Acoustic and Vibration ,Visual and Scene ,Motion and Position ,Force and Tactile Optical, Electromagnetic and Radiation Environmental, Biological and Chemical Other Signals ,Processors for Edge AI ,Edge AI Hardware Architecture ,Microcontrollers and Digital Signal Processors System-on-Chip ,Deep Learning Accelerators ,FPGAs and ASICs ,Edge Servers ,Multi-Device Architectures ,Devices and Workloads. Textbook 1: Chapter 3	

Module-3	9 Hours
Edge AI in the Real World: Common Use Cases for Edge AI, Greenfield and Brown-field Projects, Real-World Products ,Types of Applications ,Keeping Track of Objects Understanding and Controlling Systems, Understanding People and Living Things Transforming Signals, Building Applications Responsibly ,Responsible Design and AI Ethics, Black Boxes and Bias ,Technology That Harms, Not Helps. Textbook 1: Chapter 2	
Module-4	9 Hours
The “Hello World” of TinyML: Building and Training a Model: What We’re Building Our Machine Learning Toolchain,Python and Jupyter Notebooks, Google Colab-oratory ,TensorFlow and Keras, Building Our Model ,Importing Dependencies, Generating Data Splitting the Data, Defining a Basic Model ,Training Our Model, Training Metrics, Graphing the History, Improving Our Model, Testing, Converting the Model for Tensor-Flow Lite, Converting to a C File. Textbook 2: Chapter 4	
Module-5	9 Hours
The “Hello World” of TinyML: Building an Application: Walking Through the Tests ,Including Dependencies, Setting Up the Test, Getting Ready to Log Data, Mapping Our Model, Creating an AllOpsResolver, Defining a Tensor Arena, Creating an Interpreter Inspecting the Input Tensor, Running Inference on an Input, Reading the Output, Running the Tests, Project File Structure, Walking Through the Source, Starting with main.functions.cc, Handling Output with output_handler.cc, Wrapping Up main.functions.cc, Understanding main.cc, Running Our Application. Textbook 2: Chapter 5	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Describe the concepts of Edge AI, embedded systems, IoT and TinyML and their importance in modern AI applications.
CO2	Explain different sensors, signals, processors and hardware architectures used in Edge AI systems.
CO3	Evaluate real-world Edge AI use cases and evaluate ethical considerations in AI system design.
CO4	Demonstrate basic machine learning models using TensorFlow and Python for TinyML applications.
CO5	Analyze TinyML applications on embedded platforms, including model conversion and inference execution.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in components A, Band C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Daniel Situnayake, Jenny Plunkett, AI at the Edge: Solving Real-World Problems with Embedded Machine Learning, O'Reilly Media, Sebastopol, CA, USA, 1st Edition, 2023.
2. Pete Warden, Daniel Situnayake, TinyML: Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers, O'Reilly Media, Sebastopol, CA, USA, 1st Edition, 2020.

Web links:

1. <https://www.youtube.com/watch?v=34rL1V9PpZQ>
2. <https://www.youtube.com/watch?v=mIZPxtJ-9EY>
3. <https://www.youtube.com/watch?v=4sK8JH4bX9A>
4. <https://www.youtube.com/watch?v=ibm6ZRI6Sm4>
5. <https://www.youtube.com/watch?v=Jk3F3KqJp3A>
6. <https://www.youtube.com/watch?v=iCmlKyAp8eQ>

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	DATA GOVERNANCE AND SECURITY		
Course Code	BCS515G	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand the concepts, frameworks and importance of data governance in organizations.
2. Learn data quality management and metadata management techniques.
3. Understand the fundamentals of data security, cryptography and access control.
4. Study data privacy regulations, risk management and compliance frameworks.
5. Explore implementation strategies, governance tools and emerging trends in data governance.

Module-1

9 Hours

The Journey to Becoming Data-Driven: Recent Technology Developments and Industry Trends, Data Management, Analytics Is Fragmenting the Data Landscape, The Speed of Software Delivery Is Changing, The Cloud's Impact on Data Management Is Immeasurable, Privacy and Security Concerns Are a Top Priority, Operational and Analytical Systems Need to Be Integrated, Organizations Operate in Collaborative Ecosystems, Enterprises Are Saddled with Outdated Data Architectures, Defining a Data Strategy, Wrapping Up

Organizing Data Using Data Domains: Application Design Starting Points, Inspirations from Software Architecture, Data Domains, Principles for Distributed and Domain-Oriented Data Management, Transitioning Toward Distributed and Domain-Oriented Data Management, Wrapping Up

Textbook: Chapter 1, Chapter 2

Module-2

9 Hours

Mapping Domains To A Technology Architecture: Domain Topologies: Managing Problem Spaces, Landing Zone Topologies: Managing Solution Spaces, Wrapping Up

Data Product Management: What Are Data Products?, Data Product Design Patterns, Design Principles for Data Products, Data Product Architecture, Solution Design, Getting Started, Wrapping Up

Textbook: Chapter 3, Chapter 4

Module-3	9 Hours
Services And API Management: Introducing API Management, What Is Service-Oriented Architecture?, A Modern View of API Management, Microservices, Ecosystem Communication, Experience APIs, Practical Example, Metadata Management, Read-Oriented APIs Serving Data Products, Wrapping Up Event And Notification Management: Introduction to Events, What Do Modern Event-Driven Architectures Look Like?, Streaming as the Operational Backbone, Guarantees and Consistency, Governance and Self-Service, Wrapping Up Textbook: Chapter 5, Chapter 6	
Module-4	9 Hours
Connecting The Dots: Cross-Domain Interoperability, Inspiring, Motivating and Guiding for Change, Organizational Transformation, Wrapping Up Data Governance And Data Security: Data Governance, Data Security, Wrapping Up Democratizing Data With Metadata: Metadata Management, The Enterprise Metadata Model, The Metalake Architecture, Wrapping Up Textbook: Chapter 7, Chapter 8, Chapter 9	
Module-5	9 Hours
Modern Master Data Management: Master Data Management Styles, Data Integration, Designing a Master Data Management Solution, Domain-Oriented Master Data Management, MDM and Data Curation, Data Governance Recommendations, Wrapping Up Turning Data Into Value: The Challenges of Turning Data into Value, Domain Data Stores, Best Practices, Business Intelligence, Advanced Analytics (MLOps), Wrapping Up Putting Theory Into Practice: A Brief Reflection on Your Data Journey, Centralized or Decentralized?, Making It Real, Data-Driven Culture, The Role of Enterprise Architects, Last Words Textbook: Chapter 10, Chapter 11, Chapter 12	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the concepts, principles and frameworks of data governance.
CO2	Apply data quality and metadata management techniques for effective data management.
CO3	Analyze data security mechanisms including encryption and access control models.

CO4	Analyze privacy regulations, risk management and compliance requirements in data governance.
CO5	Implement data governance strategies and identify emerging trends in secure data management.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbook:

1. Piethein Strengholt, Data Management at Scale, 2nd Edition 2023.

Reference Books:

1. John Ladley, Data Governance: How to Design, Deploy and Sustain an Effective Data Governance Program, 2nd Edition, Morgan Kaufmann 2019.
2. Erwin, Data Governance and Data Stewardship, 1st Edition McGraw Hill .
3. Stuart McKendrick, Data Governance: The Definitive Guide, 1st Edition O'Reilly 2023.
4. William Stallings, Cryptography and Network Security: Principles and Practice, 8th Edition Pearson, 2022.

-
5. David Loshin, Master Data Management, 1st Edition, Morgan Kaufmann Publishers 2008.

Web links:

1. https://onlinecourses.nptel.ac.in/noc22_cs37/preview
2. <https://www.coursera.org/learn/data-privacy-security-governance-risk-and-compliance>
3. <https://datagovernance.com/>

Self Study Topics:

1. Data Governance Frameworks, Data Quality Management Tools, Data Privacy Regulations (GDPR / DPDP Act), Data Catalog and Metadata Tools, Cloud Data Governance, AI Governance and Ethical Data Usage.

Activity Based Learning:

1. Study and present a case study on data privacy regulations such as GDPR or India's Digital Personal Data Protection Act.
2. Perform a data quality assessment on a sample dataset and identify issues related to accuracy, completeness and consistency.

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	SITE RELIABILITY ENGINEERING		
Course Code	BCS515H	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Introduce the principles and culture of Site Reliability Engineering (SRE) in large-scale systems.
2. Understand service reliability metrics such as SLIs, SLOs and error budgets.
3. Learn operational practices such as monitoring, incident management and troubleshooting.
4. Apply reliability engineering techniques including automation, load balancing and chaos engineering.

Module-1	9 Hours
Introduction: The Sysadmin Approach to Service Management, Tenets of SRE. The Production Environment at Google, from the Viewpoint of an SRE: Hardware, System Software that Organizes the Hardware. Embracing Risk: Managing Risk, Measuring Service Risk, Risk tolerance of Service. Textbook: Chapter 1, Chapter 2, Chapter 3	
Module-2	9 Hours
Service Level Objectives: Service Level Terminology, Indicators in Practice, Objectives in Practice. Eliminating Toil: Toil Defined, Why Less Toil is Better. Monitoring Distributed Systems: The Four Golden Signals, Black-Box Versus White-Box. Textbook: Chapter 4, Chapter 5, Chapter 6	
Module-3	9 Hours
The Evolution of Automation at Google: The Value of Automation, The Use Cases for Automation. Release Engineering: The Role of a Release Engineer, Philosophy, Continuous Build and Deployment.	

Simplicity: System Stability Versus Agility, Minimal APIs.

Textbook: Chapter 7, Chapter 8, Chapter 9

Module-4

9 Hours

Practical Alerting from Time-Series Data: Alerting, Sharding the Monitoring Topology, Black-Box Monitoring.

Being On-Call: Life of an On-Call Engineer, Balanced On-Call, Avoiding Inappropriate Operational Load.

Effective Troubleshooting: Theory of Troubleshooting, Making Troubleshooting Easier.

Textbook: Chapter 10, Chapter 11, Chapter 12

Module-5

9 Hours

Emergency Response: Change-Induced Emergency, Process-Induced Emergency, Learn from the Past. Don't Repeat It.

Managing Incidents: The Anatomy of an Unmanaged Incident, Elements of Incident Management Process, A Managed Incident.

Postmortem Culture: Learning from Failure: Introducing a Postmortem Culture.

Textbook: Chapter 13, Chapter 14, Chapter 15

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the fundamentals of Site Reliability Engineering (SRE), its evolution, principles, roles and service lifecycle in large-scale production environments.
CO2	Apply core SRE principles such as SLIs, SLOs, SLAs, error budgets, monitoring and automation to manage reliability and operational efficiency.
CO3	Analyze operational practices including incident management, alerting, troubleshooting and postmortem analysis to improve system reliability.
CO4	Analyze reliability engineering techniques such as chaos engineering, load balancing, failure handling and distributed system management.
CO5	Implement SRE organizational practices including team structure, on-call management and collaboration models for effective service operations.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Niall Richard Murphy, Betsy Beyer, Chris Jones, Jennifer Petoff, Site Reliability Engineering: How Google Runs Production Systems, 1st Edition, O'Reilly Media 2016.

Reference Books:

1. Betsy Beyer et al., The Site Reliability Workbook, 1st Edition, O'Reilly Media, 2018.
2. Brendan Burns, Designing Distributed Systems, 1st Edition, O'Reilly Media, 2018.
3. Martin Kleppmann, Designing Data-Intensive Applications, 1st Edition, O'Reilly Media, 2017.
4. Thomas A. Limoncelli, The Practice of Cloud System Administration, 1st Edition, Addison-Wesley 2014.

Web links:

1. <https://sre.google/sre-book/>

-
2. <https://coursera.org/learn/developing-a-google-sre-culture/>

Self Learning:

1. Certification Courses (NPTEL / Coursera / Infosys SpringBoard / Udemy)



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	API DEVELOPMENT AND MICROSERVICES		
Course Code	BCS515I	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand fundamental concepts and paradigms of API development
2. Learn API design principles and best practices
3. Analyze techniques for scaling and evolving APIs
4. Understand developer ecosystem and API product strategy
5. Gain knowledge of microservices architecture and infrastructure

Module-1	9 Hours
What's an API: Why Do We Need APIs, Who Are Our Users, The Business Case for APIs, What Makes an API Great. API Paradigms: Request-Response APIs, Event-Driven APIs. API Security: Authentication and Authorization, OAuth, WebHooks Security API. Design Practices: Designing for Real-Life Use Cases, Designing for a Great Developer Experience. Textbook 1: Chapter 1, Chapter 2, Chapter 3, Chapter 4	
Module-2	9 Hours
Scaling APIs: Scaling Throughput, Evolving API Design, Pagination techniques, Rate Limiting APIs, Developer SDKs. Managing Change: Toward Consistency, Backward Compatibility, Planning and Communicating Changes. Textbook 1: Chapter 6, Chapter 7	
Module-3	9 Hours
Building a Developer Ecosystem Strategy: Building a Developer Strategy, Deriving Measurements. Developer Resources: API Documentation Samples and Snippet, Software Development Kits and Frameworks, Development Tools, Rich Media, Community Contribution.	

Developer Programs: Defining Your Developer Programs, Deep Developer Programs, Broad Developer Programs.

Textbook 1: Chapter 8, Chapter 9, Chapter 10

Module-4

9 Hours

Toward a Microservices Architecture: What Are Microservices? The Coordination Cost Problem, The Hard Parts, The “Up and Running” Microservices Model.

Designing a Microservices Operating Model: Team Size and Team Skills, Inter-team Coordination, Introduction to Team Topologies, Team Types, Interaction Modes. Designing a Microservices Team Topology- Establish a System Design Team, Building a Microservices Team Template, Platform Teams, Enabling and Complicated-Subsystem Teams, Consumer Teams.

Textbook 2: Chapter 1, Chapter 2

Module-5

9 Hours

Building a Microservices Infrastructure: Infrastructure Components: The Network, The Kubernetes Service, The GitOps Deployment Server. Implementing the Infrastructure - Installing kubectl, Setting Up the Module Repositories, The Network Module, The Kubernetes Module, Setting Up Argo CD, Testing the Environment, Cleaning Up the Infrastructure.

Textbook 2: Chapter 7

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain API concepts, paradigms and design principles.
CO2	Apply techniques for designing scalable and maintainable APIs.
CO3	Analyze API lifecycle management and developer ecosystem strategies.
CO4	Analyze and design microservices architecture and team structures.
CO5	Implement microservices infrastructure using modern tools like Kubernetes and GitOps.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.

Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE Evaluation is done for a maximum of 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Brenda Jin, Saurabh Sahni, Designing Web APIs, 1st Edition, O'Reilly Publications, 2018.
2. Ronnie Mitra and Irakli Nadareishvili, Microservices: Up and Running: A Step-by-Step Guide to Building a Microservices Architecture, 1st Edition, O'Reilly Media, 2020.

Reference Books:

1. Mark Masse, REST API Design Rulebook, 1st Edition, O'Reilly Media, 2011.
2. Chris Richardson, Microservices Patterns: With Examples in Java, 1st Edition, Manning Publications, 2018.

Web links:

1. <https://www.coursera.org/learn/applications-development-microservices-serverless-openshift>
2. <https://www.udemy.com/topic/microservices/>
3. <https://www.scholarhat.com/free-course/net-microservices-course>

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	INTRODUCTION TO COMPUTER FORENSICS		
Course Code	BCS515J	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand the fundamental concepts of computer forensics and its applications in law enforcement, military and business environments.
2. Explain cybercrime occurrences, forensic services and risk management techniques used in investigating and preventing cyber crimes.
3. Apply techniques for data recovery, data backup and digital evidence acquisition in computer forensic investigations.
4. Apply procedures for the collection, preservation, duplication and analysis of digital evidence while ensuring integrity and maintaining proper chain of custody.

Module-1	9 Hours
Computer Forensics Technology: What is Computer forensics?, Use of Computer Forensics in Law Enforcement, Computer Forensics Assistance to Human Resources / Employment Proceedings, Computer Forensics Services, Benefits of Professional Forensics Methodology. Types of Computer Forensics Technology: Types of Military Computer Forensic Technology, Types of Law Enforcement Computer Forensic Technology, Types of Business Computer Forensic Technology. Textbook: Chapter 1 (1.1 - 1.5), Chapter 2 (2.1 - 2.3)	
Module-2	9 Hours
Types of Vendor and Computer Forensics Services: Occurrence of Cyber Crime, Cyber Detectives, Fighting Cyber Crime with Risk Management Techniques, Computer Forensics Investigative Services, Forensic Process Improvement. Computer Forensics Evidence and Capture: Data Recovery Defined, Data Backup and Recovery, The Role of Backup in Data Recovery, The Data-Recovery Solution. Textbook: Chapter 3 (3.1 - 3.5), Chapter 4 (4.1 - 4.4)	

Module-3	9 Hours
Evidence Collection and Data Seizure: Why Collect Evidence?, Collection Options, Obstacles, Types of Evidence, The Rules of Evidence, Volatile Evidence, General Procedure, Collection and Archiving, Methods of Collection, Artifacts, Collection Steps, Controlling Contamination: The Chain of Custody. Textbook: Chapter 5 (5.1 - 5.12)	
Module-4	9 Hours
Duplication and Preservation of Digital Evidence: Preserving the Digital Crime Scene, Computer Evidence Processing Steps, Legal Aspects of Collecting and Preserving Computer Forensic Evidence. Computer Image Verification and Authentication: Special Needs of Evidential Authentication, Practical Consideration, Practical Implementation. Textbook: Chapter 6 (6.1 - 6.3), Chapter 7 (7.1 - 7.3)	
Module-5	9 Hours
Computer Forensics Analysis: Discovery of Electronic Evidence - Electronic Document Discovery: A Powerful New Litigation Tool. Reconstructing Past Events: How to Become a Digital Detective, Usable File Formats, Unusable File Formats, Converting Files. Networks: Network Forensics Scenario, Technical Approach, Destruction of E-Mail, Damaging Computer Evidence, Documenting the Intrusion or Destruction of Data, System Testing. Textbook: Chapter 8 (8.1), Chapter 10 (10.1 - 10.4), Chapter 11 (11.1 - 11.6)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the fundamental concepts of computer forensics, its technologies and its applications in law enforcement, military and business environments.
CO2	Identify different vendor services, cyber crime occurrences and risk management techniques used in computer forensic investigations.
CO3	Apply appropriate techniques for data recovery, data backup and digital evidence capture in forensic investigations.
CO4	Demonstrate the procedures for evidence collection, data seizure, preservation, duplication and maintaining the chain of custody in digital forensic investigations.
CO5	Analyze electronic evidence, network intrusions and digital artifacts to reconstruct events and support forensic investigation and litigation processes.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in components A, Band C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. John R. Vacca, Computer Forensics: Computer Crime Scene Investigation, 2nd Edition, Charles River Media, 2005.

Reference Books:

1. Christof Paar, Jan Pelzl, Understanding Cryptography: A Textbook for Students and Practitioners, 2nd Edition, Springer, 2010.
2. Ali Jahangiri, Live Hacking: The Ultimate Guide to Hacking Techniques & Countermeasures for Ethical Hackers & IT Security Experts, 1st Edition, 2009.

Weblinks:

1. https://onlinecourses.nptel.ac.in/noc26_lw01/preview
2. <https://11nq.com/gf1628q>

Self Study Topics:

1. Explore Autopsy Digital Forensics Tool.

-
2. Analyze Network Traffic Using Wireshark.
 3. Study Password Cracking Techniques Using John the Ripper.

Activity Based Learning:

1. Certifications (Skill based): Industry recognized certifications (NPTEL, CISCO, Infosys Springboard).



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	BIOLOGY FOR ENGINEERS		
Course Code	BBOE507	CIE MARKS	50
(L : T : P) + SL	(2:0:0)+(30 Hours/Sem)	SEE MARKS	50
Credits	2	EXAM HOURS	2

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Familiarize students with basic biological concepts and their engineering applications.
2. Enable students to understand bio-design principles for creating novel devices and structures.
3. Provide an understanding of how biological systems can be redesigned as substitute products for natural systems.
4. Motivate students to develop an interdisciplinary vision of biological engineering.

Module-1	6 Hours
Introduction to Fundamentals of Life: Structure and functions of a cell. Biomolecules: Properties and functions of carbohydrates, nucleic acids, proteins, enzymes and lipids.	
Module-2	6 Hours
Applications of Biomolecules: Carbohydrates in cellulose-based water filters production, PHA and PLA in bioplastics production, nucleic acids in vaccines and diagnosis, proteins in food production, enzymes in biosensors fabrication and food processing, enzymes in biobleaching, lipids in biodiesel and detergents production.	
Module-3	6 Hours
Bioengineering Design Inspired by Anatomy: Brain as a CPU system, eye as a camera system, heart as a pump system, lungs as a purification system, kidney as a filtration system.	
Module-4	6 Hours
Nature-Inspired Biomaterials and Mechanisms: Echolocation, photosynthesis, bird flying, lotus leaf effect, plant burrs, shark skin, kingfisher beak. Artificial Blood: Haemoglobin-based oxygen carriers (HBOCs) and perfluorocarbons (PFCs).	

Module-5

6 Hours

Trends in Bioengineering: Tissue engineering, stem cells and their applications, bio-printing techniques and materials.

Electrical tongue and electrical nose in food science, DNA origami and biocomputing, bioimaging and artificial intelligence for disease diagnosis, bioconcrete.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Corroborate the concepts of biomimetics for specific requirements.
CO2	Understand basic biological concepts for relevant industrial applications.
CO3	Evaluate design and development principles to apply engineering knowledge in biological systems.
CO4	Think critically to explore innovative bio-based solutions for socially relevant problems.

ASSESSMENT AND EVALUATION REGULATIONS OF BSC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Average of Two Internal Assessment Tests (MCQs) each conducted for 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two assessment methods as per 24RNBE4.2 scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation is done for a maximum of 50 marks.
SEE (C)	50	18	Question papers having 50 MCQ with each question carrying 01 marks and Exam duration shall be 2 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, Band C.

Textbooks:

1. Chandrashekar B.M., Basavaraju B.C., Basuchandra's Biology for Engineers, 1st Edition, 2025.
2. Rajendra Singh C and Rathnakar Rao N, Biology for Engineers, Rajendra Singh C and Rathnakar Rao N Publishing, 1st Edition, 2023.

Reference Books:

1. Stuart Ira Fox and Krista Rompolski, Human Physiology, 16th Edition, McGraw-Hill LLC, 2022.
2. S. Thyagarajan, N. Selvamurugan, M.P. Rajesh, R.A. Nazeer, Richard W. Thilagaraj, S. Barathi and M.K. Jaganathan, Biology for Engineers, 1st Edition, McGraw Hill Education (India) Private Limited, New Delhi, 2014.
3. Arthur T. Johnson, Biology for Engineers, 1st Edition, CRC Press (Taylor & Francis Group), 2011.
4. Leslie Cromwell, Fred J. Weibell and Erich A. Pfeiffer, Biomedical Instrumentation and Measurements, 2nd Edition, Prentice Hall (Pearson India Education), 2011.
5. Sohini Singh and Tanu Allen, Biology for Engineers, 1st Edition, Vayu Education of India, New Delhi, 2014.
6. Yoseph Bar-Cohen, Biomimetics: Nature-Based Innovation, 1st Edition, CRC Press (Taylor & Francis Group), 2011.
7. Dario Floreano and Claudio Mattiussi, Bio-Inspired Artificial Intelligence: Theories, Methods and Technologies, 1st Edition, The MIT Press, 2008.
8. C.R. Sunilkumar, N. Geetha and A.C. Udayashankar, Bioremediation of Heavy Metals: Bacterial Participation, 1st Edition, LAP Lambert Academic Publishing, 2019.

ESTD: 2001

An Institute with a Difference

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	RESEARCH METHODOLOGY AND IPR		
Course Code	BRMK557	CIE MARKS	50
(L : T : P) + SL	(2:0:0)+(30 Hours/Sem)	SEE MARKS	50
Credits	2	EXAM HOURS	2

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Explain the fundamental concepts, objectives and types of research in engineering.
2. Apply literature review techniques, technical reading strategies and appropriate attribution and citation practices in academic research.
3. Demonstrate ethical practices and identify research misconduct in engineering research.
4. Explain the concepts and significance of Intellectual Property Rights (IPR) in engineering.
5. Understand the procedures for protecting intellectual property through patents, copyrights, trademarks, industrial designs and geographical indications.

Module-1	6 Hours
Introduction: Meaning of Research, Objectives of Engineering Research and Motivation in Engineering Research, Types of Engineering Research, Finding and Solving a Worthwhile Problem. Ethics in Engineering Research, Ethics in Engineering Research Practice, Types of Research Misconduct, Ethical Issues Related to Authorship. Textbook: Chapter 1 (1.1 - 1.9)	
Module-2	6 Hours
Literature Review and Technical Reading: Introduction to literature review and technical reading, understanding new and existing knowledge and techniques for analysis and synthesis of prior art. Overview of bibliographic databases such as Web of Science, Google and Google Scholar. Effective search strategies and approaches for technical reading. Conceptualizing research through critical and creative reading practices Attributions and Citations: Importance of giving proper credit through attributions and citations. Functions and attributes of citations and the role of titles and keywords in enhancing citation impact. Understanding knowledge flow through citations and the practice of citing datasets. Overview of citation styles, along with guidelines for acknowledgments and proper attribution in academic writing Textbook: Chapter 2 (2.1 - 2.5) Chapter 3 (3.1 - 3.9)	

Module-3	6 Hours
Introduction To Intellectual Property: Role of IP in the Economic and Cultural Development of the Society, IP Governance, IP as a Global Indicator of Innovation, Origin of IP History of IP in India. Major Amendments in IP Laws and Acts in India. Patents: Conditions for Obtaining a Patent Protection, To Patent or Not to Patent an Invention. Rights Associated with Patents. Enforcement of Patent Rights. Inventions Eligible for Patenting. Non-Patentable Matters. Patent Infringements. Avoid Public Disclosure of an Invention before Patenting. Process of Patenting Textbook: Chapter 4 (4.1 - 4.6) Chapter 5 (5.1 - 5.8)	
Module-4	6 Hours
Process of Patenting: Prior Art Search. Choice of Application to be Filed. Patent Application Forms. Jurisdiction of Filing Patent Application. Publication. Pre-grant Opposition. Examination. Grant of a Patent. Validity of Patent Protection. Post-grant Opposition. Commercialization of a Patent. Need for a Patent Attorney/Agent. Can a Worldwide Patent be Obtained? Do I Need First to File a Patent in India? Patent Related Forms. Fee Structure Copyrights and Related Rights: Classes of Copyrights. Criteria for Copyright. Ownership of Copyright. Copyrights of the Author. Copyright Infringements. Copyright Infringement is a Criminal Offence. Copyright Infringement is a Cognizable Offence. Fair Use Doctrine. Copyrights and Internet. Non-Copyright Work. Copyright Registration. Judicial Powers of the Registrar of Copyrights. Fee Structure. Copyright Symbol. Validity of Copyright. Textbook: Chapter 6 (6.1 - 6.19) Chapter 7 (7.1 - 7.16)	
Module-5	6 Hours
Trademarks: Eligibility Criteria. Who Can Apply for a Trademark. Acts and Laws. Designation of Trademark Symbols. Classification of Trademarks. Registration of a Trademark is Not Compulsory. Validity of Trademark. Types of Trademark Registered in India. Famous Case Law: Coca-Cola Company vs. Bisleri International Pvt. Ltd. Industrial Designs: Eligibility Criteria. Acts and Laws to Govern Industrial Designs. Design Rights. Enforcement of Design Rights. Non-Protectable Industrial Designs India. International Treaties. Famous Case Law: Apple Inc. vs. Samsung Electronics Co. Geographical Indications: Acts, Laws and Rules Pertaining to GI. Ownership of GI. Rights Granted to the Holders. Registered GI in India. Identification of Registered GI. Textbook: Chapter 8 (8.1 - 8.12) Chapter 9 (9.1 - 9.17) Chapter 10 (10.1 - 10.5)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Describe the fundamental concepts, objectives, classifications and methodologies of engineering research.
CO2	Analyze and apply concepts of literature review, technical reading, attribution and citation in research contexts.
CO3	Identify ethical issues and research misconduct and demonstrate ethical practices in engineering research.
CO4	Interpret and explain the role and significance of Intellectual Property Rights (IPR) in engineering and innovation.
CO5	Differentiate between patents, copyrights, trademarks, industrial designs and geographical indications and identify their applications in intellectual property protection.

ASSESSMENT AND EVALUATION REGULATIONS OF AEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Average of Two Internal Assessment Tests (MCQs) each conducted for 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two assessment methods as per 24RNBE4.2 scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation is done for a maximum of 50 marks.
SEE (C)	50	18	Question papers having 50 MCQ with each question carrying 01 marks and Exam duration shall be 2 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.

Textbooks:

1. Dr. Santosh M Nejar and Dr. Harish Bendigeri, Research Methodology and Intellectual Property Rights, 1st Edition, Published by the Authors, 2023.

Reference Books:

1. David V. Thiel, Research Methods for Engineers, 1st Edition, Cambridge University Press, 2014.
2. N.K. Acharya, Intellectual Property Rights, 6th Edition, Asia Law House, 2014.



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	NATIONAL SERVICE SCHEME		
Course Code	BNSK559	CIE MARKS	100
(L : T : P)	(0:0:2)	SEE MARKS	—
Credits	—	EXAM HOURS	—

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Identify the needs and problems of the community and involve in problem solving.
2. Develop a sense of social and civic responsibility and utilize knowledge in finding practical solutions to individual and community problems.
3. Develop competence required for group living and sharing of responsibilities and gain skills in mobilising community participation to acquire leadership qualities and democratic attitudes.

Module-1

Waste management– Public, Private and Govt organization, 5 R's.

Module-2

Automation of local Activities

Module-3

Organic farming, Indian Agriculture (Past, Present and Future), Connectivity for marketing.

Module-4

Spreading public awareness under rural outreach Programs.

Module-5

Food preservation and Packaging

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Understand the motto and concepts of NSS.
CO2	Develop the overall personality of volunteers and make them responsible citizens and leaders of the nation.
CO3	Analyze the environmental and societal problems/issues and will be able to design solutions for the same
CO4	Analyze the environmental and societal problems/issues and will be able to design solutions for the same.

ASSESSMENT AND EVALUATION REGULATIONS OF MC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE - IA Tests (A)	60	-	Sum of Two Internal Assessment Tests (MCQs) each conducted for 30 marks.
Continuous and Comprehensive Evaluation (B)	40	-	Any two Assessment methods as per 24RNBE4.2, scaled to 40 marks.
Total CIE (A+B)	100	40	CIE Evaluation is done for a maximum of 100 marks.
The minimum marks to be secured in CIE is 40 (40%) out of allotted 100 marks to Pass.			

Reference Books:

1. NSS Course Manual, Published by NSS Cell, VTU Belagavi.
2. Government of Karnataka, NSS Cell, Activities Reports and its Manual.
3. Government of India, NSS Cell, Activities Reports and its Manual.

Web Links:

1. <https://nss.gov.in/>
2. <https://pledge.mygov.in/>

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	PHYSICAL EDUCATION		
Course Code	BPEK559	CIE MARKS	100
(L : T : P)	(0:0:2)	SEE MARKS	—
Credits	—	EXAM HOURS	—

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand the fundamental concepts of Physical Education, games and sports skills, Health, Nutrition and Fitness
2. Familiarization of health-related Exercises, Sports for overall growth and development
3. Participate in the competition at regional/state / national / international levels.
4. Create consciousness among the students on Health, Fitness and Wellness in developing and maintaining a healthy lifestyle.

Module-1 Title: Athletics-Track Events-110 and 400 Mtrs Hurdles 3 Hours

110 Mtrs and 400Mtrs:

1. Hurdling Technique: Lead leg Technique, Trail leg Technique
2. Crouch start (its variations) use of Starting Block.

Module-2 Title: Athletics - Jumps 3 Hours

High Jump

- a) Approach Run
- b) Take-off
- c) Bar Clearance (Straddle)
- d) Landing

Module-3 Title: Athletics - Throws 3 Hours

Discus Throw

- a) Holding the Discus
- b) Initial Stance Primary Swing
- c) Rotation
- d) Release and Recovery (Rotation in the circle)

Module-4 Title: Football	3 Hours
I. Fundamental skills a) Kicking: Kicking the ball with inside of the foot, Kicking the ball with Full Instep of the foot, Kicking the ball with Inner Instep of the foot, Kicking the ball with Outer Instep of the foot and Lofted Kick. b) Trapping: Trapping the Rolling ball and the Bouncing ball with sole of the foot. c) Dribbling: Dribbling the ball with Instep of the foot, Dribbling the ball with Inner and Outer Instep of the foot. d) Heading: In standing, running and jumping condition. e) Throw-in: Standing throw-in and Running throw-in. f) Tackling: Simple Tackling, Slide Tackling. g) Goal Keeping: Collection of Ball, Ball clearance- kicking, throwing and deflecting. h) Game practice with application of Rules and Regulations II. Rules and their interpretation and duties of the officials	
Module-5 Title: Hockey	3 Hours
I. Fundamental skills a) Passing: Short pass, Long pass, push pass, hit b) Trapping. c) Dribbling and Dozing. d) Penalty stroke practice. e) Penalty corner practice. f) Tackling: Simple Tackling, Slide Tackling. g) Goal Keeping, Ball clearance- kicking and deflecting. h) Game practice with application of Rules and Regulations II. Rules and their interpretation and duties of the officials	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Understand and demonstrate fundamental techniques of track events including 110m and 400m hurdles, with proper use of starting blocks and hurdling skills.
CO2	Apply technical knowledge of athletic jumps, particularly high jump, including approach run, take-off, bar clearance and landing techniques.

CO3	Develop skills and perform throwing events, specifically discus throw, with correct techniques such as grip, stance, rotation, release and recovery.
CO4	Demonstrate fundamental skills of football such as kicking, trapping, dribbling, heading, throw-in, tackling and goalkeeping along with understanding rules and officiating.
CO5	Exhibit basic skills and techniques in hockey, including passing, trapping, dribbling, tackling, penalty strokes/corners and understand rules and duties of officials.

ASSESSMENT AND EVALUATION REGULATIONS OF MC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	60	-	Sum of Two Internal Assessment Tests (MCQs) each conducted for 30 marks.
Project (B)	40	-	—
Total CIE (A+B)	100	40	CIE Evaluation is done for a maximum of 100 marks.
The minimum marks to be secured in CIE is 40 (40%) out of 100 marks allotted to Pass.			

Textbooks:

1. Health and Physical Education by V.K. Sharma
2. Introduction to Physical Education, Fitness and Sport by Daryl Siedentop
3. Physical Education Activities, Sports and Games: The Origin and Background, Rules of Play, Teaching Hints and Fundamentals by Louis E. Means

Reference Books:

1. Saraswati Health and Physical Education Textbook
2. Foundations of Physical Education, Exercise Science and Sport by Deborah A. Wuest

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – V

Course Title	YOGA		
Course Code	BYOK559	CIE MARKS	100
(L : T : P)	(0:0:2)	SEE MARKS	—
Credits	—	EXAM HOURS	—

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Physical, Mental and Spiritual objectives of Yoga
2. Explain the meaning of Yoga
3. Discuss the importance of Yoga and its scope
4. Describe the aims and objectives of Yoga
5. Clarify the misconceptions on Yoga

Module-1 Title: Suryanamaskara		3 Hours
Suryanamaskar 12 count 6 rounds 1. Suryanamaskar prayer. 2. Meaning, need & Importance 3. Benefits of Suryanamaskara		
Module-2 Title: Yoga		3 Hours
1. Meaning and definition. 2. Brief introduction of yogic practices for the common man 3. Rules and regulations 4. Misconceptions of Yoga		
Module-3 Title: Asanas - Sitting & Standing		3 Hours
Need and importance of different Asanas, meaning by name, technique, precautionary measures and benefits of each asana. Sitting 1. Ardha Ushtrasana 2. Yogamudra in Padmasana		

Standing

1. Hastapadasana
2. Utkatasana

Module-4 Title: Asanas - Supine line

3 Hours

Need and importance of different Asanas, meaning by name, technique, precautionary measures and benefits of each asana.

Supine line

1. Sarvangasana
2. Chakrasana

Module-5 Title: Pranayama

3 Hours

1. Kapalabhati - Revision of practice 40 strokes/min - 3 rounds
2. Pranayama:
 - a. Ujjayi
 - b. Sheetali
3. Meaning, Need, importance of Pranayama. Different types. Meaning by name, technique, precautionary measures and benefits of each Pranayama

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the meaning, definition, scope and objectives of Yoga and its role in physical, mental and spiritual well-being.
CO2	Demonstrate Suryanamaskara (12 counts, 6 rounds) with correct technique and understand its importance and benefits.
CO3	Apply knowledge of yogic practices, rules and regulations while identifying and correcting common misconceptions of Yoga.
CO4	Perform sitting, standing and supine Asanas with proper technique, precautions and awareness of their benefits.
CO5	Practice Pranayama techniques (Kapalabhati, Ujjayi, Sheetali) to improve breathing control, concentration and overall health.

ASSESSMENT AND EVALUATION REGULATIONS OF MC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	60	-	Sum of Two Internal Assessment Tests (MCQs) each conducted for 30 marks.
Project (B)	40	—	—
Total CIE (A+B)	100	40	CIE Evaluation is done for a maximum of 100 marks.
The minimum marks to be secured in CIE is 40 (40%) out of 100 marks allotted to Pass.			

Textbooks:

1. The Yoga Sutras of Patanjali by Sri Swami Satchidananda.
2. Light on Yoga by B.K.S. Iyengar.

Reference Books:

1. The Heart of Yoga by T.K.V. Desikachar.
2. Asana Pranayama Mudra Bandha by Swami Satyananda Saraswati.

VI SEMESTER SUBJECTS

ESTD: 2001
An Institute with a Difference

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	CLOUD COMPUTING		
Course Code	BCS601	CIE MARKS	50
(L : T : P) + SL	(3:0:2)+(45 Hours/Sem)	SEE MARKS	50
Credits	4	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Introduce the fundamental concepts, architectures and system models of distributed and cloud computing, focusing on scalability, performance, security and energy efficiency.
2. Provide knowledge of virtualization technologies and mechanisms, including virtual machines, CPU/memory/I/O virtualization, virtual clusters and data center automation.
3. Expose students to Hadoop and MapReduce frameworks for large-scale data processing and cloud application design methodologies.
4. Enable students to design and develop cloud-based applications using Python, including data-intensive and analytics-driven applications.
5. Familiarize students with cloud security concepts and practices, including identity and access management, authentication, authorization, data security, key management and auditing.

Module–1	9 Hours
Distributed System Models and Enabling Technologies: Scalable Computing Over the Internet, Technologies for Network Based Systems, System Models for Distributed and Cloud Computing, Software Environments for Distributed Systems and Clouds, Performance, Security and Energy Efficiency. Textbook 1: Chapter 1 (1.1 - 1.5)	
Module–2	9 Hours
Virtual Machines and Virtualization of Clusters and Data Centers: Implementation Levels of Virtualization, Virtualization Structure/ Tools and Mechanisms, Virtualization of CPU/Memory and I/O devices, Virtual Clusters and Resource Management, Virtualization for Data Center Automation Textbook 1: Chapter 3 (3.1 - 3.5)	

Module–3	9 Hours
Hadoop & MapReduce: Apache Hadoop, Hadoop MapReduce Job Execution, Hadoop Schedulers, Hadoop Cluster Setup Cloud Application Design: Design consideration for cloud applications, Cloud Application Design Methodologies, Data Storage Approaches Textbook 2: Chapter 4 (4.1 - 4.4,) Chapter 5 (5.2 - 5.5)	
Module–4	9 Hours
Cloud Application Development in Python: Design Approaches, Image Processing App, Document Storage App, MapReduce App, Social Media Analytics App Textbook 2: Chapter 8 (8.1 - 8.5)	
Module–5	9 Hours
Cloud Security: Introduction, CSA Cloud security Architecture, Authentication, Authorization, Identity & Access Management, Data Security, Key Management, Auditing Textbook 2: Chapter 12 (12.1 - 12.8)	

PRACTICAL COMPONENT OF IPCC

EXPERIMENTS
Program 1: Virtualization & VM Setup Scenario: A startup company wants to test its software on multiple operating systems without purchasing separate physical machines. Question: As a system engineer, you are asked to install a virtualization tool and create a virtual machine (VM). Configure the VM with appropriate CPU cores, memory, storage and network settings. Demonstrate successful VM creation by verifying OS boot-up and network connectivity (e.g., ping or internet access). Application: Used for software testing across multiple OS platforms, sandboxing untrusted apps, running legacy applications and setting up dev/test environments without buying extra hardware.
Program 2: Resource Monitoring in Virtualized Environment Scenario: An IT administrator wants to analyze how virtual machines utilize system resources to avoid performance bottlenecks.

Question:

Develop a program to monitor CPU usage, memory utilization, disk space and network statistics of a virtual machine. Record and display the performance metrics periodically to help administrators assess the efficiency of the virtualized environment.

Application:

Used for software testing across multiple OS platforms, sandboxing untrusted apps, running legacy applications and setting up dev/test environments without buying extra hardware.

Program 3: Virtual Cluster Creation

Scenario:

A research team requires multiple interconnected machines to simulate a distributed computing environment for experimentation.

Question:

Create a virtual cluster using multiple virtual machines or containers. Configure passwordless communication among the nodes and demonstrate resource sharing by executing commands or sharing files across the cluster.

Application:

Used to build distributed system testbeds for research/teaching, simulate clusters for Big Data / HPC / microservices and validate parallel processing + inter-node communication before deploying on real cloud infrastructure.

Program 4: Hadoop Installation (Pseudo-Distributed Mode)

Scenario:

A data analytics team wants to process large datasets using Hadoop on a single machine before moving to a full cluster.

Question:

Install and configure Apache Hadoop in pseudo-distributed mode. Execute a sample Hadoop job and verify the successful execution by checking the output stored in HDFS.

Application:

Used for local Big Data prototyping, learning Hadoop components on a single machine, validating configuration and jobs before scaling to a multi-node Hadoop cluster in industry.

Program 5: HDFS File System Operations

Scenario:

A cloud storage administrator is responsible for managing files and access permissions in a distributed file system.

Question:

Demonstrate various HDFS operations, including:

1. Creating directories
2. Uploading and downloading files
3. Setting file permissions
4. Displaying file system statistics

Explain how these operations support large-scale data storage and access control.

Application:

Used in organizations to manage large-scale distributed storage, enforce access control & permissions, support reliable storage through replication and enable shared data access for analytics teams.

Program 6: MapReduce – Word Count Application**Scenario:**

A publishing company wants to analyze the frequency of words in large text documents to identify commonly used terms.

Question:

Implement a MapReduce program using Hadoop to count the frequency of words in a given text file. Analyze the job execution process, including the roles of the Mapper, Reducer and HDFS.

Application:

Used for text analytics such as finding trending keywords, indexing documents, log key-word frequency analysis and building features for search engines, NLP pipelines and content mining.

Program 7: MapReduce – Large Dataset Analysis**Scenario:**

An e-commerce company needs insights from massive transaction or log files to understand customer behavior.

Question:

Develop a MapReduce program to analyze large datasets such as log files or transaction data. Generate meaningful summary statistics such as frequency counts or top-N results and explain how MapReduce enables scalable data processing.

Application:

Used for log analytics, clickstream processing, transaction summarization, fraud pattern aggregation and generating top-N insights (top products/users/errors) from massive datasets.

Program 8: Cloud-Based Document Storage Application

Scenario:

A company wants a secure cloud solution for employees to store and manage official documents online.

Question:

Design and implement a cloud-based document storage application using Python that supports:

1. Uploading documents
2. Storing documents securely
3. Retrieving documents
4. Deleting documents

Explain how cloud storage improves accessibility and reliability.

Application:

Used for enterprise document management (HR, finance, academic records), secure file sharing, remote access, backup/versioning and centralized storage for collaboration.

Program 9: Cloud-Based Image Processing Application

Scenario:

A media company needs an online service to process large numbers of images efficiently.

Question:

Develop a cloud-based image processing application using Python that performs operations such as image resizing, format conversion, or grayscale transformation. Demonstrate how cloud computing improves scalability and performance.

Application:

Used in media and e-commerce for bulk image resizing/compression, thumbnail generation, format conversions, watermarking and scalable processing pipelines (e.g., product catalogs).

Program 10: Cloud Security Mechanisms

Scenario:

A cloud service provider must ensure that user data is protected against unauthorized access and security threats.

Question: Implement cloud security mechanisms for a cloud-based application, including:

1. Authentication
2. Authorization
3. Data encryption
4. Auditing

Explain how each mechanism contributes to secure cloud computing.

Application:

Used to protect cloud apps by ensuring only valid users access data, preventing breaches, meeting compliance (audit trails), securing sensitive data via encryption and supporting secure multi-user cloud services.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Interpret the concepts, architectures and system models of distributed and cloud computing.
CO2	Analyze virtualization techniques and tools, CPU, memory and I/O virtualization, virtual clusters and data center automation in cloud environments.
CO3	Apply Hadoop and MapReduce frameworks to process and analyze large-scale data and perform distributed computing tasks.
CO4	Construct cloud-based applications using Python, incorporating appropriate cloud application design methodologies and data storage approaches.
CO5	Develop cloud security solutions incorporating authentication, authorization, identity and access management, data security, key management and auditing.

ASSESSMENT AND EVALUATION REGULATIONS OF IPCC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	20	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 20 marks.
Continuous and Comprehensive Evaluation (B)	10	-	Any one Assessment methods as per 24RNBE4.2 scaled to 10 marks.
Total CIE Theory (A+B)	30	12	Marks of Tests and Assignments awarded for a maximum of 30.

CIE Practical	12	-	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE Practical Test	8	-	Average of marks of Two tests, each conducted for 100 marks covering all experiments, scaled to 8.
Total CIE Practical (C)	20	8	Marks of Experiments, Record and Test awarded for a maximum of 20 marks.
CIE: Theory + Practical (A+B+C)	50	20	CIE Evaluation is done for a maximum of 50 marks.
SEE (D)	50	18	SEE is a theory exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall be 3 hours.
CIE+SEE (A+B+C+D)	100	40	Final marks shall be the sum of marks scored in Components A, B, C and D.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE. The laboratory component of the IPCC shall be for CIE only. However, in SEE, the questions from the laboratory component shall be included in their respective modules only.			

Textbooks:

1. Kai Hwang, Geoffrey C. Fox and Jack J. Dongarra, Distributed and Cloud Computing: From Parallel Processing to the Internet of Things, 1st Edition, Morgan Kaufmann (Elsevier), 2012.
2. Arshdeep Bahga and Vijay Madisetti, Cloud Computing: A Hands-on Approach, 1st Edition, Universities Press, 2014.

Reference Books:

1. Rajkumar Buyya, Christian Vecchiola, Thamrai Selvi, Shivananda Poojara and Satish N Srirama. Mastering Cloud Computing, McGraw-Hill Education, 2nd Edition, 2024.
2. Toby Velte and Anthony Velte, Cloud Computing: A Practical Approach, McGraw-Hill Education, 1st edition, 2009.
3. George Reese, Cloud Application Architectures: Building Applications and Infrastructure in the Cloud, O'Reilly Publication, 1st Edition, 2009.

4. John Rhoton, Cloud Computing Explained: Implementation Handbook for Enterprises, Recursive Press, 2nd Edition, 2009.

Web Links:

1. <https://www.ibm.com/topics/cloud-computing>
2. <https://www.redhat.com/en/topics/virtualization>
3. <https://hadoop.apache.org/docs/.../MapReduceTutorial.html>
4. <https://realpython.com/python-cloud-development/>
5. <https://cloud.google.com/security/best-practices>

Activity Based Learning:

1. Cloud Architecture Case Study Discussion
2. Students design a virtual cluster architecture for a given application scenario (e.g., e-commerce, analytics).
3. Students design Map and Reduce logic for real-life problems: Word frequency, Log analysis, Sales data aggregation
4. Hands-on coding of mini cloud applications: Document storage, Image processing
Students Analyze social media datasets

Self Study Topics:

1. Cloud Computing Courses in Google Cloud Career Launchpad program.

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING		
Course Code	BCS602	CIE MARKS	50
(L : T : P) + SL	(3:2:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	4	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Explain the basics of Artificial Intelligence & Machine Learning algorithms
2. Identify the problems applicable for Artificial Intelligence and Machine Learning techniques
3. Discuss knowledge representation issues and different kind of learning algorithms
4. Demonstrate application of algorithms to real-world problems

Module-1		9 Hours
Introduction: What is AI? The state of the art. Intelligent agents: Agents and environment, Good Behavior: The Concept of Rationality, The nature of Environments, The structure of Agents. Textbook 1: Chapter 1 (1.1, 1.4), Chapter 2 (2.1 - 2.4)		
Module-2		9 Hours
Problem Solving by Searching: Problem-solving agents, Example Problems, Search algorithms, Uninformed Search Strategies, Informed search Strategies. Adversarial search: Game Theory, Optimal Decision in games. Textbook 1: Chapter 3 (3.1 - 3.5), Chapter 6 (6.1 - 6.2)		
Module-3		9 Hours
Introduction: Need for Machine Learning, Machine Learning Explained, Machine Learning in Relation to other Fields, Types of Machine Learning, Challenges of Machine Learning, Machine Learning Process, Machine Learning Applications. Similarity-based Learning: Nearest-Neighbor Learning, Weighted K-Nearest-Neighbor Algorithm, Nearest Centroid Classifier, Locally Weighted Regression (LWR). Regression Analysis: Introduction to Regression, Introduction to Linearity correlation and Causation, Introduction to Linear Regression, Multiple Linear Regression, Polynomial Regression, Logistic Regression. Textbook 2: Chapter 1, Chapter 4 (4.2 - 4.5), Chapter 5 (5.1 - 5.3, 5.5 - 5.7)		

Module-4	9 Hours
Bayesian Learning: Introduction to Probability-based Learning, Fundamentals of Bayes Theorem, Classification Using Bayes Model, Naïve Bayes Algorithm for Continuous Attributes. Artificial Neural Networks: Introduction, Biological Neurons, Artificial Neurons, Perceptron and Learning Theory, Types of Artificial Neural Networks, Popular Applications of Artificial Neural Networks, Advantages and Disadvantages of ANN, Challenges of ANN. Textbook 2: Chapter 8 (8.1 - 8.4), Chapter 10 (10.1 - 10.5, 10.9 - 10.11)	
Module-5	9 Hours
Decision Tree Learning: Introduction to Decision Tree Learning Model, Decision Tree Induction Algorithms. Reinforcement Learning: Overview of Reinforcement Learning, Scope of Reinforcement Learning, Reinforcement Learning as Machine Learning, Components of Reinforcement Learning, Model Free Methods, Q-Learning. Textbook 2: Chapter 6 (6.1 - 6.2), Chapter 14 (14.1 - 14.4, 14.8, 14.9)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Understand how intelligent agents work in real-world applications.
CO2	Utilize search algorithms to construct effective AI agent solutions.
CO3	Examine machine learning techniques to address real-world problems.
CO4	Implement Bayesian learning methods for solving classification tasks.
CO5	Demonstrate machine learning applications using neural networks and reinforcement learning approaches.

ASSESSMENT AND EVALUATION REGULATIONS OF PCC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.

Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, 4th Edition, Pearson Education, 2023.
2. S Sridhar, M Vijayalakshmi, Machine Learning, Oxford University Press, 2021, First Edition.

Reference Books:

1. Aurelien Geron, Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools and Techniques to Build Intelligent Systems, 3rd Edition, Shroff / O'Reilly Media, 2022.
2. Tom M. Mitchell, Machine Learning, 1st Edition, McGraw Hill Education, 2017.
3. Ethem Alpaydin, Introduction to Machine Learning, 2nd Edition, PHI Learning Pvt. Ltd., 2014.
4. George F. Luger, Structures and Strategies for Complex Problem Solving, 5th Edition, Pearson Education, 2011.
5. Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements of Statistical Learning, 2nd Edition, Springer New York, NY, 2009.

Web Links:

1. <https://nptel.ac.in/courses/106106139>
2. <https://nptel.ac.in/courses/106102220>
3. <https://www.coursera.org/.../machine-learning-introduction>

Self Study Topics:

1. Compare Uninformed vs Informed Search.

-
2. Implement BFS & DFS in Python.
 3. Explore A* Search with heuristics.
 4. Learn Bayes' theorem applications in classification.
 5. Naive Bayes text classification example.
 6. Neural Networks and Backpropagation demo.
 7. Q-Learning on a grid world.
 8. Temporal Difference vs Monte Carlo Method.
 9. Utility-based Agents vs Goal-based Agents.

Activity Based Learning:

1. Certifications (Skill-Based) – Industry-recognized certifications (e.g., CISCO, NPTEL, Infosys Springboard).



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	QUANTUM COMPUTING		
Course Code	BCS613A	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The objectives of the course are to facilitate the learners to:

1. Explain the fundamental concepts of classical information theory and the mathematical representation of quantum states using qubits and the Bloch sphere.
2. Apply matrix operations, quantum postulates and tensor products to calculate and describe the evolution and interaction of single and multi-qubit systems.
3. Construct quantum gates and circuits to execute standard communication protocols such as quantum teleportation and superdense coding.
4. Compute the performance metrics of key quantum algorithms and implement introductory error correction techniques to mitigate the impact of environmental noise on quantum information.

Module-1

9 Hours

Classical Information Theory: Information content in a signal, Entropy and Shannon's Information Theory. Probability Basics.

Qubits and Quantum States: The Qubit as a fundamental unit, Vector spaces, Linear combinations of vectors, Basis and Dimension. Inner products, Orthonormality, Gram-Schmidt Orthogonalization, Bra-Ket formalism. The Cauchy-Schwartz and Triangle Inequalities in quantum states.

Textbook 1: Chapter 1 (1.1 - 1.4), Chapter 2 (2.1 - 2.10)

Module-2

9 Hours

Matrices and Operators: The Pauli Operators (X, Y, Z), Outer Products, The Closure Relation. Representation of operators using matrices, Matrix representation of operators in two-dimensional spaces, Definition of Pauli Matrices. Hermitian, Unitary and Normal Operators. Eigenvalues, Eigenvectors, the Characteristic Equation, Spectral Decomposition. The Trace of an Operator, Important Properties of the Trace, The Expectation Value of an Operator, Postulates of Quantum Mechanics: The State of a system, Observables, Measurements and Time Evolution of the system.

Textbook 1: Chapter 3 (3.1 - 3.13, 3.19)

Module-3	9 Hours
Tensor Products: Representing composite states in Quantum Mechanics, Computing inner products, Tensor products of column vector, Tensor products of matrices. The Density Operator: The Density Operator for a Pure state, The Density Operator for a Mixed state, Key properties of the Density operator, Characterizing mixed states, The Partial Trace and the Reduced Density Operator, The Density Operator and the Bloch Vector. Measurement Theory: Projective measurements, Measurements on composite system and Positive Operator-Valued Measures (POVM). Textbook 1: Chapter 4 (4.1 - 4.3, 4.5), Chapter 5 (5.1 - 5.6), Chapter 6 (6.1 - 6.4)	
Module-4	9 Hours
Quantum Entanglement: Bell's Theorem, Bipartite systems and the Bell Basis, Entanglement Fidelity, Schmidt Decomposition. Quantum Gates and Circuits: Classical logic gates, Single-Qubit gates (X, Y, Z, Hadamard, Phase-shift gates), Basic circuit diagrams, Controlled gates (CNOT, Controlled-Z) and Gate Decomposition. Applications of Entanglement: Quantum Teleportation and Superdense Coding. Textbook 1: Chapter 7 (7.1 - 7.2, 7.5, 7.7), Chapter 8 (8.1 - 8.2, 8.6 - 8.8), Chapter 10 (10.1 - 10.4)	
Module-5	9 Hours
Quantum Algorithms: Quantum Parallelism, Deutsch-Jozsa Algorithm, Shor's Algorithm, Quantum Searching and Grover's Algorithm. Quantum Noise and Error Correction: Single-qubit errors, Quantum Operations and Kraus Operators, Depolarization channel, Bit flip and Phase flip Channels, Amplitude Damping, Phase damping, Quantum Error Correction. Textbook 1: Chapter 9 (9.5 - 9.6, 9.9, 9.10), Chapter 12 (12.1 - 12.7)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Analyse the core principles of classical information theory alongside the mathematical foundations underpinning quantum states and qubits.
CO2	Implement matrix algebra and quantum postulates to represent and manipulate quantum operators and observables.
CO3	Evaluate tensor products and density operator formalism for representing composite quantum systems, distinguishing between pure and mixed states.

CO4	Apply entanglement principles and quantum gates to design basic quantum circuits and assess communication protocols.
CO5	Assess standard quantum algorithms and error correction techniques to address quantum computing challenges.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. David McMahon, Quantum Computing Explained, 1st Edition, Wiley-Interscience (IEEE Computer Society), 2008.

Reference Books:

1. Michael A. Nielsen and Isaac L. Chuang, Quantum Computation and Quantum Information, 10th Anniversary Edition, Cambridge University Press, 2010.
2. Phillip Kaye, Raymond Laflamme and Michele Mosca, An Introduction to Quantum Computing, 1st Edition, Oxford University Press, 2007.

Web links:

1. <https://nptel.ac.in/courses/106106232>

Self Study Topics:

1. Comparative Analysis of Physical Qubit Implementations: Investigating the architectural trade-offs between superconducting circuits, trapped and photonic systems regarding coherence times and scalability.
2. Quantum Circuit Design and Simulation: Developing hands-on proficiency in using frameworks like Qiskit or Cirq to implement single-qubit and multi-qubit gates on simulated and real quantum hardware.
3. Quantum vs. Classical Error Correction Paradigms: Evaluating how the No-Cloning Theorem necessitates unique quantum strategies, such as Shor's 9-qubit code, to mitigate bit-flip and phase-flip noise.
4. Hybrid Quantum-Classical Algorithms: Exploring the integration of classical CPU processing with quantum coprocessors for variational algorithms and optimization tasks.

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	MULTI-AGENT SYSTEMS		
Course Code	BCS613B	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. To introduce the concept of a multi agent systems and Distributed Constraints.
2. Explore the main issues surrounding the computer and extended form games.
3. Develop cooperative learning, stochastic games.
4. Exhibit the awareness about protocols about multi agent resource allocation and auctions.
5. Construct voting mechanism design.

Module-1	9 Hours
Multiagent Problem Formulation: Utility is Not Money, Expected Utility, Multiagent Markov Decision Processes, Partially Observable MDPs, Hierarchical Planning. Distributed Constraint Satisfaction: Defining distributed constraint satisfaction problems, Domain-pruning algorithms, Heuristic search algorithms, History and references. Distributed Constraints: Filtering Algorithm, Hyper-Resolution Based Consistency Algorithm, Asynchronous Backtracking, Asynchronous Weak-Commitment Search, Distributed Breakout, Distributed Constraint Optimization: Adopt, OptAPO. Textbook 1: Chapter 1 (1.1 - 1.3), Chapter 2 Textbook 2: Chapter 1	
Module-2	9 Hours
Standard and Extended Form Games: Games in Normal Form, Games in Extended Form, Finding a Solution. Characteristic Form Games and Coalition Formation: Characteristic Form Games, Coalition Formation. Introduction to Non-cooperative Game Theory: Games in Normal Form: Self-interested agents. Textbook 1: Chapter 3, Chapter 4 Textbook 2: Chapter 3 (3.1)	

Module-3	9 Hours
Learning in Multiagent Systems: The Machine Learning Problem, Cooperative Learning, Repeated Games, Stochastic Games, General Theories for Learning Agents, Collective Intelligence. Textbook 1: Chapter 5 (5.1 - 5.6)	
Module-4	9 Hours
Negotiation: The Bargaining Problem, Monotonic Concession Protocol, Negotiation as Distributed Search, Ad-hoc Negotiation Strategies, The Task Allocation Problem, Complex Deals, Argumentation-Based Negotiation, Negotiation Networks. Textbook 1: Chapter 6 (6.1 - 6.8)	
Module-5	9 Hours
Auctions: Valuations, Simple Auctions, Combinatorial Auctions. Voting and Mechanism Design: The Voting Problem, Mechanism Design, Mechanism Design Summary. Nature-Inspired Approaches: Ants and Termites, Immune System, Physics. Textbook 1: Chapter 7, Chapter 8, Chapter 10	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Apply decision-making processes under different constraints.
CO2	Analyze games in different forms.
CO3	Apply cooperative learning techniques in game development.
CO4	Analyze negotiation strategies in Multi-Agent Systems.
CO5	Apply voting methods to solve decision-making problems in Multi-Agent Systems.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.

Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. José M. Vidal, Fundamentals of Multiagent Systems, 1st Edition, Published independently by the Author, 2006.
2. Yoav Shoham and Kevin Leyton-Brown, Multiagent Systems: Algorithmic, Game-Theoretic and Logical Foundations, 1st Edition, Cambridge University Press, 2008.

Reference Books:

1. Gerhard Weiss (Ed.), Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence, 1st Edition, The MIT Press, 1999.

Web Links:

1. <https://nptel.ac.in/courses/106/105/106105077/>
2. <https://slink.com/nigoaz0>
3. <https://www.youtube.com/watch?v=02su1u2AXG0>
4. https://www.youtube.com/watch?v=gUrENDkPw_k
5. https://youtu.be/gUrENDkPw_k?si=0hAszkVVPXM-j0sT
6. <https://www.youtube.com/watch?v=lPxFn1-mhVY>
7. <https://www.youtube.com/watch?v=E1ifAXKDy0I>
8. <https://www.youtube.com/watch?v=yi7doi-QGJI>

Self Study Topics:

1. Deep Reinforcement Learning in multiagent systems.
2. Federated learning with multiagents.
3. Multiagent systems in IoT environments.
4. Ethical issues in autonomous agents.



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	AUGMENTED REALITY AND VIRTUAL REALITY		
Course Code	BCS613C	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Provide students with a strong foundation in the fundamental concepts, principles and technologies of Virtual Reality (VR) and Augmented Reality (AR).
2. Familiarize students with VR/AR hardware components, software frameworks, input-output devices and human-computer interaction aspects.
3. Enable students to understand modeling techniques, interaction methods and sensory systems used in immersive virtual environments.
4. Develop the ability to design, analyze and implement basic VR/AR applications for real-world problem solving.

Module-1	9 Hours
Introduction To Virtule Reality (VR): Introduction to Virtual Reality, Modern VR Experiences, Telepresence, History Repeats. Bird's-Eye View: Hardware, Software, Human Physiology and Perception. Textbook 1: Chapter 1 (1.1 - 1.3), Chapter 2 (2.1 - 2.3)	
Module-2	9 Hours
Input Devices (Trackers, Navigation and Gesture Interfaces): Three-dimensional position trackers, Tracker Performance Parameters, Electromagnetic Trackers, Trackers Using AC and DC Magnetic Fields, Navigation and Manipulation Interfaces, Gesture Interfaces. Output Devices: Graphics Displays, Sound Displays and Haptic Feedback. Textbook 2: Chapter 2 (2.1 - 2.1.2.3), (2.2 - 2.2.2.2), (2.3 - 2.3.2.1), Chapter 3 (3.2 - 3.3.2), Chapter 4 (4.1 - 4.1.2.1), (4.2 - 4.2.1.2)	
Module-3	9 Hours
Modeling: Geometric Modeling – Virtual Object Shape, Kinematics Modeling, Physical Modeling, Behaviour Modeling, Model Management. Textbook 2: Chapter 6 (6.1 - 6.1.1.4), (6.2 - 6.2.5), (6.3 - 6.3.2.2), (6.4 - 6.5.1.4)	

Module-4	9 Hours
Augmented Reality (AR): Introduction to Augmented Reality, Difference between Augmented Reality and Virtual Reality, How Augmented Reality Works with Technology, Hardware Components to Power Augmented Reality, Augmented Reality Business Applications, Tools Available for Augmented Reality and Recognition. Textbook 3: Chapter 1 (1.1 - 1.5)	
Module-5	9 Hours
Interaction and Audio: Interaction – Motor Programs and Remapping, Locomotion, Manipulation, Social Interaction. Audio – The Physics of Sound, The Physiology of Human Hearing, Auditory Perception, Auditory Rendering. Textbook 1: Chapter 10 (10.1 - 10.4), Chapter 11 (11.1 - 11.4)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the concepts of Virtual Reality and communication media.
CO2	Explain the hardware and software components of Virtual Reality systems.
CO3	Explain various Virtual Reality modeling approaches.
CO4	Analyze human factors and applications of Virtual Reality.
CO5	Develop Virtual Reality applications using appropriate development tools.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.

SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Steven M. LaValle, Virtual Reality, 1st Edition, Cambridge University Press, 2020.
2. Grigore C. Burdea and Philippe Coiffet, Virtual Reality Technology, 3rd Edition, Wiley-IEEE Press, 2024.
3. P. Kaliraj and T. Devi, Innovating with Augmented Reality: Applications in Education and Industry, 1st Edition, Auerbach Publications (CRC Press), 2021.

Reference Books:

1. Rajesh K. Maurya, Computer Graphics with Virtual Reality Systems, 3rd Edition, Wiley India Pvt. Ltd., 2018.
2. William R. Sherman and Alan B. Craig, Understanding Virtual Reality: Interface, Application and Design, 2nd Edition, Morgan Kaufmann (Elsevier), 2018.
3. Grigore C. Burdea and Philippe Coiffet, Virtual Reality Technology, 2nd Edition, John Wiley & Sons, Inc., 2003.
4. Kelly S. Hale and Kay M. Stanney, Handbook on Virtual Environments: Design, Implementation and Applications, 2nd Edition, CRC Press (Taylor & Francis Group), 2014.

Web Links:

1. <http://vr.cs.uiuc.edu/vrbook.pdf>
2. <https://nptel.ac.in/courses/106/106/106106138/>

Self Study:

Certification Courses (NPTEL / Coursera / Infosys SpringBoard / Udemy)

1. Case Studies: Analysis of real-world AR/VR applications in gaming, education labs and medical surgery training.

-
2. UX Design Assignment: Design an immersive prototype (Virtual Class, Meditation World, Historical Reconstruction, Virtual Museum) using Figma / Canva / sketches.
 3. VR Interaction Study: Teleportation, gaze interaction and controller-based interaction.



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	CRYPTOGRAPHY AND NETWORK SECURITY		
Course Code	BCS613D	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand the basics of Cryptography concepts, Security and its principle.
2. Analyse different Cryptographic Algorithms.
3. Illustrate public and private key cryptography.
4. Understand the key distribution scenario and certification.
5. Understand approaches and techniques to build protection mechanism in order to secure computer networks.

Module-1

9 Hours

Information and Network security concepts: Cyber security, Information security and network security, The OSI Security architecture, Security Attacks, Services and Mechanism, Cryptography, Network security.

Introduction to Number theory: The Euclidean Algorithm, Modular arithmetic, Prime numbers, Fermat's and Euler's theorems, Testing of primality, Discrete logarithms.

Textbook: Chapter 1 (1.1 - 1.8), Chapter 2 (2.2 - 2.6, 2.8)

Module-2

9 Hours

Classical encryption techniques: Symmetric cipher model, Substitution Techniques, Transposition techniques.

Block Ciphers and Data Encryption Standard: Traditional Block Cipher structures, data Encryption Standard (DES), A DES Example, The strength of DES, Block cipher design principles.

Textbook: Chapter 3 (3.1 - 3.3), Chapter 4 (4.1 - 4.5)

Module-3

9 Hours

Public key cryptography and RSA: Principles of public key cryptosystems, The RSA algorithm: Description of the Algorithm, Computational aspects, The Security of RSA. Other public key crypto systems: Diffie-Hellman key exchange.

Message Authentication Codes: Message Authentication Requirements, Message Authentication Functions, Requirement for Message Authentication Codes.

Digital Signatures: Digital Signatures, NIST Digital Signature Algorithm.	
Textbook: Chapter 9 (9.1 - 9.2), Chapter 10 (10.1), Chapter 12 (12.1 - 12.3), Chapter 13 (13.1, 13.4)	
Module-4	9 Hours
Cryptographic Key Management and Distribution: Distribution of Public keys, X.509 certificates, Public key infrastructure.	
User Authentication: Kerberos.	
Transport layer security: Web security consideration, Transport layer security.	
Textbook: Chapter 15 (15.3 - 15.5), Chapter 16 (16.3), Chapter 17 (17.1 - 17.2)	
Module-5	9 Hours
Email security: Email Threats and comprehensive email security, S/MIME, DNSSEC, DomainKeys identified Mail.	
IP Security: IP Security overview, IP Security Policy, Encapsulating Security Payload.	
Network endpoint security: Firewalls, Intrusion detection system, Malicious Software.	
Textbook: Chapter 19 (19.3 - 19.5, 19.8), Chapter 20 (20.1 - 20.3), Chapter 21 (21.1 - 21.3)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain computer and network security concepts and number-theoretic principles used in cryptography.
CO2	Apply classical symmetric encryption techniques and block ciphers for secure communication.
CO3	Analyze public key cryptographic algorithms, message authentication codes and digital signatures.
CO4	Apply key management, public key infrastructure and authentication protocols for secure email communication.
CO5	Analyze security threats and countermeasures using IPSec, intrusion detection systems, malware defenses and firewalls.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in components A, Band C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. William Stallings, Cryptography and Network Security: Principles and Practice, 8th Edition, Pearson Education, 2020.

Reference Books:

1. Behrouz A. Forouzan and Debdeep Mukhopadhyay, Cryptography and Network Security, 3rd Edition (Special Indian Edition), McGraw Hill Education (India) Private Limited, 2015.
2. Keith M. Martin, Everyday Cryptography: Fundamental Principles and Applications, 1st Edition, Oxford University Press, 2012.
3. V.K. Pachghare, Cryptography and Information Security, 2nd Edition, PHI Learning Private Limited, 2015.

Web links:

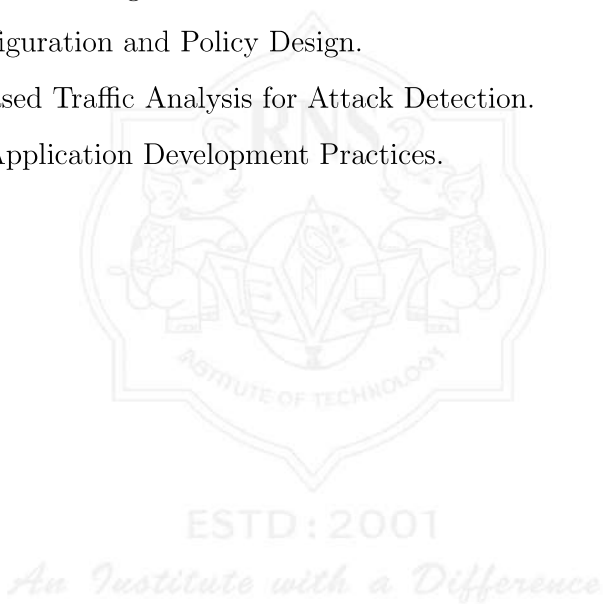
1. <https://nptel.ac.in/courses/106105162>

Self Study Topics:

1. Finite fields: Groups, Rings, Fields.
2. Advanced Encryption Standards.
3. Elliptic curve cryptography, Pseudorandom Number Generation Based on an Asymmetric Cipher.

Activity Based Learning:

1. Attack Simulation Using Network Simulators.
2. Firewall Configuration and Policy Design.
3. Wireshark-Based Traffic Analysis for Attack Detection.
4. Secure Web Application Development Practices.



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	SECURE PROGRAMMING		
Course Code	BCS613E	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Introduce the basic concepts and goals of secure coding.
2. Understand common software vulnerabilities.
3. Learn secure coding techniques to prevent memory and input-related issues.
4. Familiarize with secure error handling, logging and web security practices.

Module-1	9 Hours
Security Concepts: Security Policy, Security Flaws, Vulnerabilities. The Security Development Life Cycle: TSP-Secure, Planning and Tracking, Quality Management. Software Security problems: Static analysis in the big picture, Classification of Vulnerabilities. Introduction to Static Analysis: Solving problems with static analysis: Type Checking, Style Checking, Program understanding, verifications and property checking, Bug finding, Security Review. Textbook 1: Chapter 1 (1.2), Chapter 9 (9.1) Textbook 2: Chapter 1 (1.4 - 1.5), Chapter 2 (2.1 - 2.2)	
Module-2	9 Hours
Strings: Common String manipulation error - Improperly bounded string copies, Off by one errors, Null termination errors, String truncation, String errors without functions, String Vulnerabilities and Exploits: Trained data, security flaw, Buffer overflow, Process Memory organization, Stack smashing, Code injection, Arc injection, Return-oriented programming, Mitigation Strategies for strings, String handling functions. Textbook 1: Chapter 2 (2.2 - 2.5)	
Module-3	9 Hours
Integer Security: Integer Conversions, Converting integers, Integer conversion Rank, Integer promotions, Usual arithmetic conversions, Conversions from unsigned integer types, Conversions from signed integer types, Conversion implications, Integer Operations	

Assignment, Addition, Subtraction, Multiplication, Division and remainder, shifts, Integer Vulnerabilities - Vulnerabilities, Wraparound, Conversion and truncation errors, Nonexceptional integer logic errors.

Textbook 1: Chapter 5 (5.3 - 5.5)

Module-4

9 Hours

Handling Inputs: What to validate – Validate all input, Validate input from all sources, How to validate – Use strong input validation, Avoid blacklisting, Don't mistake usability for security, Reject bad data, Make good input validation the default, Check input length, Bound numeric input.

Buffer Overflow: Introduction, Exploiting buffer overflow vulnerabilities, Buffer allocation strategies, Tracking buffer sizes

Textbook 2: Chapter 5 (5.1 - 5.2), Chapter 6 (6.1)

Module-5

9 Hours

Web Applications: Input and Output Validation for the Web: Expect That the Browser Has Been Subverted, Protect the browser from malicious content, HTTP Considerations: Use POST, Not GET, Request Ordering, Error Handling, Request Provenance, Maintaining Session State: Use Strong Session Identifiers, Enforce a Session Idle Timeout and a Maximum Session Lifetime, Begin a New Session upon Authentication, Using the Struts frameworks for input validation: Setting up the struts validator, Use the Struts validator for all actions, Validate every parameter, Maintain the validation logic.

Textbook 2: Chapter 9 (9.1 - 9.4)

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the fundamental concepts and principles of software security.
CO2	Analyze common software vulnerabilities and their impact on software systems.
CO3	Apply techniques to detect and prevent security flaws in software programs.
CO4	Implement secure coding practices in software development.
CO5	Apply secure design principles for web applications, including request handling and session management.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE Evaluation is done for a maximum of 50 marks
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE+SEE(A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Robert C. Seacord, Secure Coding in C and C++, 2nd Edition, Addison-Wesley Professional, 2013.
2. Brian Chess and Jacob West, Secure Programming with Static Analysis, 1st Edition, Addison-Wesley Professional, 2007.

Reference Books:

1. Dafydd Stuttard and Marcus Pinto, The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws, 2nd Edition, John Wiley & Sons, Inc., 2011.
2. Michael Howard and David LeBlanc, Writing Secure Code, 2nd Edition, Microsoft Press, 2003.

Web links:

1. <https://nptel.ac.in/courses/106106199>

2. <https://www.youtube.com/watch?v=c70x38RhtxA>
3. <https://www.youtube.com/watch?v=8m1N2t-WANc>
4. <https://s1lnk.com/8elb8q5>

Self Learning Activities:

1. Investigate the differences between traditional software development and secure development practices such as the Security Development Lifecycle and TSP-Secure. Analyze trade-offs in terms of security, cost and development time.
2. Gain practical experience by analyzing programs using static analysis tools like SonarQube or similar tools. Compare manual code review with automated bug detection for identifying vulnerabilities such as null pointer issues, memory leaks and insecure coding patterns.
3. Investigate how vulnerabilities like Buffer Overflow occur and study their exploitation techniques including stack smashing and code injection.
4. Develop small programs to demonstrate incorrect behavior due to improper conversions and arithmetic operations and evaluate their impact on system security.
5. Analyze common web vulnerabilities such as Cross-Site Scripting and session hijacking. Explore how improper request handling and weak session management lead to security issues and study preventive measures.

ESTD: 2001

An Institute with a Difference

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	DESIGN PATTERNS AND CLEAN CODE		
Course Code	BCS613F	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand reusable object-oriented software design
2. Learn commonly used design patterns in software systems
3. Understand principles of writing readable and maintainable code
4. Improve software quality using clean coding practices

Module-1	9 Hours
Introduction to Design Patterns: what is a design pattern, Describing Design Patterns, the catalog of design patterns, organizing the catalog, how design patterns solve design problems, how to select a design pattern, how to use a design pattern.	
Textbook 1: Chapter 1 (1.1, 1.3 - 1.8)	
Module-2	9 Hours
Creational Design Patterns: Abstract Factory Pattern, Builder Pattern, Factory Method Pattern, Singleton Pattern	
Text Book 1: Chapter 3 (3.1 - 3.3, 3.5)	
Module-3	9 Hours
Structural Design Patterns: Adapter Pattern, Composite Pattern, Decorator Pattern, Facade Pattern.	
Textbook 1: Chapter 4 (4.1, 4.3 - 4.5)	
Module-4	9 Hours
Behavioral Design Patterns: Command Pattern, Observer Pattern, Strategy Pattern, Template Method Pattern.	
Textbook 1: Chapter 5 (5.2, 5.7 - 5.9)	
Module-5	9 Hours
Clean Code Principles: Clean Code, There Will Be Code Bad Code, The Total Cost of Owning a Mess.	

Meaningful Names Introduction: Use Intention-Revealing Names, Avoid Disinformation, Make Meaningful Distinctions, Use Pronounceable Names, Use Searchable Names.

Textbook 2: Chapter 1 (1.1 - 1.3), Chapter 2 (2.1 - 2.6)

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the fundamental concepts of object-oriented design and the role of design patterns in software development.
CO2	Classify creational, structural and behavioral design patterns used in object-oriented systems.
CO3	Apply appropriate design patterns to solve common software design problems.
CO4	Analyze software code to identify code smells and design issues using refactoring techniques.
CO5	Implement clean code principles and best coding practices to develop maintainable and readable software.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in components A, Band C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, 1st Edition, Addison-Wesley Professional, 1994.
2. Robert C. Martin, Clean Code: A Handbook of Agile Software Craftsmanship, 1st Edition, Prentice Hall, 2008.

Reference Books:

1. Eric Freeman and Elisabeth Robson, Head First Design Patterns: Building Extensible and Maintainable Object-Oriented Software, 2nd Edition, O'Reilly Media, 2020.
2. Martin Fowler, Refactoring: Improving the Design of Existing Code, 2nd Edition, Addison-Wesley Professional, 2018.

Web links:

1. https://onlinecourses.nptel.ac.in/noc26_cs46/preview
2. <https://nptel.ac.in/courses/106105182>

Self Learning:

1. Certification Courses (NPTEL / Coursera / Infosys SpringBoard / Udemy)
2. <https://www.coursera.org/learn/design-patterns>
3. <https://slink.com/k190r6f>
4. <https://www.coursera.org/specializations/software-design-architecture>

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	APPLIED STATISTICAL MODELLING		
Course Code	BCS613G	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Explain the basic concepts of mathematical modelling and its applications in solving real-world engineering problems.
2. Apply mathematical techniques and computational tools to develop models for practical engineering and scientific problems.
3. Analyze different types of mathematical models and interpret system behavior using analytical and numerical methods.
4. Evaluate the accuracy and limitations of mathematical models by validating them with real data and assumptions.
5. Develop and implement mathematical models for complex real-world problems and interpret results for decision-making.

Module-1	9 Hours
Introduction The Role of Statistics in Engineering: The Engineering Method and Statistical Thinking, Collecting Engineering Data, Mechanistic and Empirical Models, Probability and Probability Methods. Probability: Sample spaces and events, Counting techniques, Interpretations and Axioms of probability , Unions of events and Addition Rules , Conditional Probability , Intersections of Events and Multiplication and Total Probability Rules , Independence , Bayes Theorem , Random Variables. Mathematical Expectations: Mean of a Random Variable , Variance and Covariance of Random Variables , Chebyshev's Theorem. Textbook 1: Chapter 1 (1.1 – 1.4) Chapter 2 (2.1 – 2.9) Textbook 2: Chapter 4 (4.1 - 4.4)	
Module-2	9 Hours
Discrete Random Variables and Probability Distribution: Probability distributions and Probability mass functions , Cumulative Distribution Functions , Mean and Variance of discrete random variables , Discrete Uniform Distribution , Binomial Distribution , Geometric and Negative Binomial Distributions , Hypergeometric Distribution , Poisson Distribution.	

Some Continuous Uniform distribution: Continuous Uniform distribution, Normal distribution, Areas under the normal curve , Applications of the normal distribution , Gamma and Exponential distributions , Chi- Squared Distributions, Beta Distribution , Lognormal Distribution.

Descriptive Statistics: Numerical Summaries of Data , Stem - and - leaf diagrams, Frequency Distributions and Histograms, Box Plots , Time Sequence Plots , Scatter Diagrams ,Probability Plots.

Textbook 1: Chapter 3 (3.1 – 3.8) Chapter 6 (6.1 - 6.7)

Textbook 2: Chapter 6 (6.1 - 6.9)

Module-3

9 Hours

Simple Linear Regression and Correlation: Introduction to Linear Regression , The Simple linear regression model , Least Squares and the Fitted Model , Properties of least squares estimators , Inferences Concerning the Regression Coefficients, Prediction , Choice of Regression Model , Analysis- of – Variance Approach , Test of linearity of regression , Data Plots and Transformations , Simple linear regression case study , correlation.

Multiple Linear Regression and Certain Nonlinear Regression Models: Introduction, Estimating the coefficients , Linear Regression Model using Matrices , Properties of Least Squares Estimators , Inferences in Multiple Regression , Choice of a fitted model through Hypothesis testing , Categorical or Indicator Variables, Sequential Methods for Model Selection.

Textbook 2: Chapter 11 (11.1 - 11.12) Chapter 12 (12.1 – 12.9)

Module-4

9 Hours

One Factor Experiments: Analysis-of-Variance Technique, The Strategy of Experimental Design, One-Way Analysis of Variance: Completely Randomized Design (One-Way ANOVA), Tests for the Equality of Several Variances, Single-Degree-of-Freedom Comparisons, Multiple Comparisons, Comparing a set of treatments in Blocks, Randomized Complete Block Designs , Graphical Methods and Model Checking, Data Transformations in Analysis of Variance, Random Effects Models , Case Study.

Design of Experiments with Several Factors: Introduction, Factorial Experiments, Two-Factor Factorial Experiments, General Factorial Experiments, 2^k Factorial Designs, Single Replicate of the 2^k Design, Addition of Center Points to a 2^k Design, Blocking and Confounding in the 2^k Design, One-Half Fraction of the 2^k Design, Smaller Fractions: The 2^{k-p} Fractional Factorial, Response Surface Methods and Design

Textbook 1: Chapter 14 (14.1 – 14.11)

Textbook 2: Chapter 13 (13.1 - 13.12)

Module-5

9 Hours

Statistical Quality Control: Quality Improvement and Statistics, Introduction to Control Charts, $\bar{X}$ and R or S Control Charts, Control Charts for Individual Measurements, Process Capability, Attribute Control Charts, Control Chart Performance, Time-Weighted Charts, Other SPC Problem-Solving Tools, Decision Theory.

Bayesian Statistics: Bayesian Concepts , Bayesian Inferences , Bayes Estimates Using Decision Theory Framework.

Textbook 1: Chapter 15 (15.1 – 15.10)

Textbook 2: Chapter 18 (18.1 – 18.3)

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the principles of mathematical modelling for real-world engineering and scientific problems.
CO2	Apply mathematical and computational techniques to formulate practical models.
CO3	Analyze mathematical models to interpret system behavior using analytical and numerical methods.
CO4	Analyze model accuracy by comparing results with real data and underlying assumptions.
CO5	Apply mathematical modelling techniques to solve complex engineering and scientific problems.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in components A, B and C.

The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.

Textbooks:

1. Douglas C. Montgomery and George C. Runger, Applied Statistics and Probability for Engineers, 7th Edition, John Wiley & Sons, Inc., 2018.
2. Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers and Keying Ye, Probability & Statistics for Engineers & Scientists, 9th Edition, Pearson Education, 2016.

Reference Books:

1. S. C. Gupta and V. K. Kapoor, Fundamentals of Mathematical Statistics, 12th Edition, Sultan Chand & Sons, 2020.

Web Links:

1. <https://www.youtube.com/watch?v=kagIBMRp2Ls&t=11s>
2. <https://www.youtube.com/watch?v=vEM4YnRFGoA>
3. <https://www.youtube.com/watch?v=BLRjywb0mes>
4. <https://www.youtube.com/watch?v=IGxPHLW6Ja4>
5. <https://www.youtube.com/watch?v=HZGCoVF3YvM>
6. <https://slink.com/9z3zwhh>
7. <https://nptel.ac.in/courses/110/105/110105093/>

Self Study Topics:

1. Markov Chains and Transition Probabilities.
2. Stochastic Processes (Poisson process, Brownian motion).
3. Law of Large Numbers and Central Limit Theorem (deep understanding).
4. Extreme Value Distributions.
5. Copula Functions (basic idea).

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	PUBLIC CLOUD ARCHITECTURE		
Course Code	BCS613H	CIE MARKS	50
(L:T:P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Recall the basic concepts of cloud computing and the features of AWS.
2. Explain the working of AWS services including EC2, S3, Glacier and EFS.
3. Illustrate the process of deploying cloud infrastructure and applications in AWS.
4. Apply AWS services to manage virtual machines and storage systems.
5. Execute database operations using DynamoDB and file-sharing using EFS.

Module-1	9 Hours
What is Amazon Web Services: What is cloud computing, What can you do with AWS, How you can benefit from using AWS, How much does it cost, Comparing alternatives, Exploring AWS services, Interacting with AWS, Creating an AWS account. A simple example: WordPress in five minutes: Creating your infrastructure, Exploring your infrastructure, How much does it cost?, Deleting your infrastructure. Textbook: Chapter 1 (1.1 - 1.8), Chapter 2 (2.1 - 2.4)	
Module-2	9 Hours
Using virtual machines-EC2: Exploring a virtual machine, Monitoring and debugging a virtual machine, Shutting down a virtual machine, Changing the size of a virtual machine, Starting a virtual machine in another data center, Allocating a public IP address, Adding an additional network interface to a virtual machine, Optimizing costs for virtual machines. Textbook: Chapter 3 (3.1 - 3.8)	
Module-3	9 Hours
Storing your objects - S3 and Glacier: What is an Object Store, Amazon S3, Backing up Data using S3 with AWS CLI, Archiving Objects to Optimize Costs, Storing Objects Programmatically, Using S3 for Static Web Hosting, Best Practices for Using S3. Textbook: Chapter 8 (8.1 - 8.7)	

Module-4	9 Hours
Sharing data volumes between machines-EFS: Creating a filesystem, Creating a mount target, Mounting the EFS share on EC2 instances, Sharing files between EC2 instances. Programming for the NoSQL database service: DynamoDB: Operating DynamoDB, DynamoDB for developers, Programming a to-do application, Creating tables, Adding data, Retrieving data, Removing data, Modifying data, Scaling capacity. Textbook: Chapter 10 (10.1 - 10.4), Chapter 13	
Module-5	9 Hours
Securing your system- IAM, security groups and VPC: Who's responsible for security?, Keeping your software up to date, Securing your AWS account, Controlling network traffic to and from your virtual machine, Creating a private network in the cloud: Amazon Virtual Private Cloud (VPC). Achieving high availability-availability zones, auto-scaling and CloudWatch: Recovering from EC2 instance failure with CloudWatch, Recovering from a data centre outage. Textbook: Chapter 6 (6.1 - 6.5), Chapter 14 (14.1, 14.2)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the fundamental concepts of cloud computing and AWS services.
CO2	Explain the functionality of compute and storage services such as EC2, S3 and Glacier.
CO3	Explain the use of EFS and DynamoDB in cloud-based applications.
CO4	Implement virtual machines and storage solutions using AWS services.
CO5	Apply AWS security and monitoring tools such as IAM, VPC and CloudWatch for basic cloud deployment.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.

Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Michael Wittig and Andreas Wittig, Amazon Web Services in Action, 2nd Edition, Manning Publications, 2018.

Reference Books:

1. Ben Piper and David Clinton, AWS Certified Solutions Architect Study Guide: Associate (SAA-C02) Exam, 3rd Edition, John Wiley & Sons, Inc., 2020.
2. Rajkumar Buyya, James Broberg and Andrzej M. Goscinski, Cloud Computing: Principles and Paradigms, 1st Edition, John Wiley & Sons, Inc., 2011.

Web links:

1. <https://www.youtube.com/playlist?list=PLdpzx00AlwvLN0xX0RfndiYSt1Le9azze>
2. <https://www.youtube.com/watch?v=k1RI5locZE4>

Self Study Topics:

1. AWS Pricing and Billing
 - 1.1 Overview of AWS pricing models
 - 1.2 Understanding billing dashboard and cost alerts
2. Amazon CloudFront (CDN)
 - 2.1 Basic concept of CDN

-
- 2.2 Delivering content using Amazon CloudFront
 - 3. AWS Elastic Load Balancer (ELB)
 - 3.1 Distributing traffic across multiple instances
 - 3.2 Types of load balancers (basic idea)
 - 4. AWS Auto Scaling
 - 4.1 Automatic scaling of resources based on demand
 - 5. Amazon RDS (Relational Database Service)
 - 5.1 Basic idea of managed databases
 - 5.2 Use cases of RDS
 - 6. Amazon DynamoDB
 - 6.1 Introduction to NoSQL database in AWS
 - 6.2 Simple use cases
 - 7. AWS Identity and Access Management (Advanced Concepts)
 - 7.1 Roles vs policies (basic understanding)
 - 7.2 Principle of least privilege
 - 8. AWS CloudWatch (Advanced Usage)
 - 8.1 Logs, metrics and alarms (practical understanding)
 - 9. AWS CloudFormation (Overview)
 - 9.1 Introduction to Infrastructure as Code in AWS
 - 9.2 Template-based resource creation (concept only)
 - 10. Basic AWS Architecture Case Study

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	INTRODUCTION TO DATA CENTER NETWORKING		
Course Code	BCS613I	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand the fundamental concepts, architecture and components of modern data center networks.
2. Explain the architecture and operational features of Junos OS used in Juniper Networks devices..
3. Configure Layer-2 networking technologies such as VLANs, link aggregation and switching mechanisms in a data center environment.
4. Analyze and configure routing protocols such as OSPF and BGP for scalable network communication.
5. Evaluate network performance and apply basic monitoring, security and automation techniques in Juniper data center networks.

Module-1	9 Hours
Fundamentals of Data Center Networking: Evolution of data center networks, Traditional vs modern data center architectures. Basic networking concepts: Switching, routing, VLANs, IP addressing. Data center components: Servers, storage, switches, routers, Three-tier architecture: Core, Aggregation, Access. SDN: Introduction to Software Defined Networking (SDN) Textbook 1: Chapter 1 (1.1 - 1.3), Chapter 2 (2.1 - 2.5), Chapter 3 (3.1 - 3.5), Chapter 7 (7.1)	
Module-2	9 Hours
Introduction to Juniper Networks and Junos OS: Overview of Juniper Networks products and solutions, Juniper hardware platforms: EX Series, QFX Series, MX Series. Architecture of Junos OS, CLI basics and configuration hierarchy, User interfaces and operational modes, Basic configuration: interfaces, hostname, user accounts Textbook 2: Chapter 1 (1.1 - 1.3), Chapter 4 (4.1 - 4.5)	

Module-3	9 Hours
Layer 2 Technologies in Juniper Data Centers: Ethernet switching concepts. VLAN configuration and trunking, Spanning Tree Protocol (STP, RSTP, MSTP), Link Aggregation (LACP), Virtual Chassis technology Layer 2 configuration in Junos Textbook 2: Chapter 3 (3.1 - 3.5), Chapter 4 (4.6 - 4.8)	
Module-4	9 Hours
Layer 3 Routing and Data Center Protocols: Static routing and dynamic routing basics, Interior Gateway Protocols, OSPF, BGP (overview for data centers). Routing configuration in Junos OS, ECMP and high availability, Introduction to data center overlay technologies (VXLAN basics) Textbook 1: Chapter 6 (6.1 - 6.6), Chapter 5 (5.1 - 5.3)	
Module-5	9 Hours
Data Center Automation, Security and Management: Network monitoring and management. Introduction to Juniper Apstra and automation concepts: Configuration backup and troubleshooting, Security basics: ACLs and firewall filters. High availability and redundancy in data centers: Case study: Modern cloud data center architecture Textbook 1: Chapter 4 (4.4 - 4.5), Chapter 7 (7.1 - 7.4), Chapter 8 (8.1 - 8.4)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the fundamentals of modern data center networking architectures and technologies.
CO2	Explain the architecture and basic configuration of Junos OS used in Juniper network devices.
CO3	Configure Layer-2 switching technologies such as VLANs, STP and link aggregation in data center environments.
CO4	Apply Layer-3 routing protocols such as OSPF and BGP in scalable data center networks.
CO5	Implement automation, monitoring and security techniques in Juniper-based data center networks.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE Evaluation is done for a maximum of 50 marks
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Dinesh G. Dutt, Cloud Native Data Center Networking: Architecture, Protocols and Tools, 1st Edition, O'Reilly Media, 2019.
2. Doug Marschke, Harry Reynolds and David Aulds, Junos Enterprise Routing: A Practical Guide to Junos Routing and Certification, 2nd Edition, O'Reilly Media, 2011.

Reference Books:

1. Mauricio Arregoces and Maurizio Portolani, Data Center Fundamentals, 1st Edition, Cisco Press, 2003.
2. Gary A. Donahue, Network Warrior, 2nd Edition, O'Reilly Media, 2011.
3. Randy Zhang and Micah Bartell, BGP Design and Implementation, 1st Edition, Cisco Press, 2003.

Web Links:

1. https://onlinecourses.nptel.ac.in/noc26_cs60/preview?utm_source=chatgpt.com
2. https://www.digimat.in/nptel/courses/video/106106243/L49.html?utm_source=chatgpt.com
3. https://www.digimat.in/nptel/courses/video/106106243/L53.html?utm_source=chatgpt.com

Self Study Topics:

1. East-West vs North-South traffic (real examples)
2. Limitations of VLAN in large-scale networks
3. Comparison: Three-tier vs Spine-Leaf architecture

Activity Based Learning

1. Certifications (Skill based): industry recognized certifications (NPTEL, CISCO, Infosys Springboard).

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	AI DRIVEN NETWORKS		
Course Code	BCS613J	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand the architecture and components of Juniper Mist AI-driven networking platform
2. Identify the AI and ML technologies powering Juniper Mist for wireless, wired and WAN operations
3. Discuss the features of Mist AI including dynamic packet capture, virtual network assistant and anomaly detection
4. Compare traditional network management with AI-driven approaches using Juniper Mist AI solutions

Module-1	9 Hours
Get Started with AI Ops: AI Native Operations Overview, Explainable AI, Requirements, AIOps in Action. Insights: Insights Overview, Site Insights, AP Insights, Organization Insights (BETA), Wireless Client Insights. Textbook: Chapter 1, Chapter 2 (2.1 - 2.5)	
Module-2	9 Hours
Service-Level Expectations (SLE): Wireless SLEs, Wired SLEs, WAN SLEs. Alerts: Alerts Overview, Juniper Mist Alert Types Textbook: Chapter 3 (3.1 - 3.3), Chapter 4 (4.1 - 4.2)	
Module-3	9 Hours
Get Started with Marvis: Marvis Virtual Network Assistant Overview, Subscriptions for Marvis. Marvis Actions: Marvis Actions Overview, Self-Driving Marvis Actions, Subscription Requirements for Marvis Actions, Layer 1 Actions. Textbook: Chapter 5, Chapter 6 (6.1 - 6.4)	

Module-4	9 Hours
Marvis Minis: Marvis Minis Overview, Marvis Minis Dashboard Overview, Add Custom URLs for Marvis Minis Validation, Exclude VLANs from Marvis Minis Validation, Disable Marvis Minis, Network and Application Monitoring with Marvis Minis, Troubleshoot Marvis Minis. Marvis Minis SLE: Marvis Minis SLE Dashboard (Beta), Network Services SLE (Marvis Minis), Application SLE (Marvis Minis). Textbook: Chapter 7, Chapter 8	
Module-5	9 Hours
Conversations and Queries: Marvis Conversations and Queries Overview, Marvis Conversational Assistant, Marvis Query Language. Marvis Client: Marvis Client Overview, Marvis Android Client, Marvis Windows Client Textbook: Chapter 9, Chapter 10(10.1 - 10.3)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Configure Mist AI-driven wireless network deployments for enterprise scenarios.
CO2	Analyze Mist AI analytics and SLE metrics to diagnose network performance issues.
CO3	Apply Marvis Virtual Network Assistant for root cause analysis and network remediation.
CO4	Implement Mist AI security policies, including access control, authentication and IoT classification.
CO5	Implement automation and management features in Mist AI for enterprise wireless networks.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.

Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in components A, Band C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Juniper Networks, Mist AI: Network Operations and Assurance, Mist AI Official Documentation and Learning Guide, Juniper Networks Inc., 2024.

Reference Books:

1. Bob Friday and Sudheer Matta, AI-Driven Networks: Mist AI in Enterprise Environments, 1st Edition, O'Reilly Media, 2023.
2. L. Andrew Coward, Brain Mechanisms: Linking Cognitive Phenomena to Neuron Activity, 1st Edition, Springer, 2017.
3. Kevin Wallace, SD-WAN Architecture and Design, 1st Edition, Cisco Press, 2022.
4. Tom M. Mitchell, Machine Learning, 1st Edition, McGraw-Hill Education, 1997.

Web links:

1. https://www.juniper.net/documentation/product/en_US/mist-systems
2. <https://learningportal.juniper.net>
3. <https://www.youtube.com/c/JuniperNetworks>
4. <https://www.juniper.net/us/en/products/cloud-services/mist-ai.html>
5. <https://api.mist.com/api/v1/docs>
6. <https://www.juniper.net/us/en/dm/mist-ai-driven-networking>
7. <https://www.juniper.net/us/en/products/mist-ai/wireless-assurance.html>
8. <https://www.juniper.net/us/en/training/certification/tracks/mist-ai.html>
9. <https://community.juniper.net>

Self Study Topics:

1. Compare traditional WLAN management with Mist AI-based assurance.
2. Explore the Marvis query interface and submit sample troubleshooting queries.
3. Study the role of Service Level Expectations (SLEs) in Wi-Fi performance monitoring.
4. Investigate Mist AI REST API using Postman for basic device queries.
5. Read about Zero Touch Provisioning for enterprise APs and switches.
6. Explore how Mist AI handles IoT device onboarding and profiling.
7. Q-learning on a grid world.
8. Review Mist AI anomaly detection mechanisms.
9. Utility-based agents vs Goal-based agents.

Activity Based Learning:

1. Certifications (Skill-Based) – industry-recognized certifications (e.g., Juniper JNCIA-MistAI, NPTEL AI courses).

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	DEVOPS AND CI/CD PIPELINES		
Course Code	BCS613K	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Introduce DevOps principles, culture and lifecycle and explain the role of DevOps tools in improving collaboration between development and operations teams.
2. Enable students to understand and use version control and build automation tools for managing source code, dependencies and software builds effectively.
3. Provide hands-on experience with Continuous Integration and Continuous Deployment tools to automate build, test and release processes.
4. Develop practical skills in configuration management and infrastructure automation using industry-standard DevOps tools.
5. Expose students to cloud-based DevOps platforms and monitoring tools to support scalable, reliable and continuous software delivery.

Module-1

9 Hours

Agile Methodology and DevOps: What is DevOps? DevOps Importance and Benefits, DevOps Principles and Practices, 7 C's of DevOps Lifecycle for Business Agility, Continuous Planning, Continuous Development, Continuous Integration, Continuous Deployment, Continuous Testing, Continuous Delivery and Monitoring, Continuous Feedback, DevOps and Continuous Testing, Reference Architecture with CI Server, How to Choose Right DevOps Tools? Steps to be Followed to Choose the Right DevOps Tools, Selecting the Right Tools, Challenges with DevOps Implementation.

Textbook: Chapter 2 (2.3 - 2.9),

Module-2

9 Hours

Map My App to DevOps Journey: Assessment, Definition, Implementation Atlassian Tools, Key Features, Competitors of Atlassian, Where can Atlassian be Best Utilized?, Pros and Cons of Atlassian, Phabricator, major components of Phabricator, Key Features, Competitors, Where can Phabricator be Best Utilized?, Pros and Cons of Phabricator, Phabricator Reference Architecture.

Textbook: Chapter 3 (3.1 - 3.3), Chapter 4 (4.1 - 4.2)

Module-3	9 Hours
Orchestration: Jenkins, Features, Example of Reference Architecture, Microsoft TFS , Features Example of Reference Architecture, TeamCity, Why Choose TeamCity as CI Server? Key Features, Popular Competitors, Where can TeamCity be Best Utilized? , Example of Reference Architecture, Ansible , Why Ansible? What can Ansible Do? Key Features, Pros and Cons, Example of Reference Architecture, Bamboo, Features, Popular Competitors, Where can Bamboo be Best Utilized? Pros and Cons, Example of Reference Architecture. Textbook: Chapter 5 (5.1 - 5.5)	
Module-4	9 Hours
Application Lifecycle Management: JIRA , Key Features, Popular Competitors of JIRA , Where can JIRA be Best Utilized?, Pros and Cons of JIRA. DevOps with Cloud, DevOps and Cloud Adoption, Benefits of using DevOps along with Cloud, Few Best Practices for DevOps in the Cloud,Reference DevOps Pipeline, DevOps in IBM Cloud – Bluemix , IBM Cloud DevOps Competitors, Pros and Cons, IBM Cloud DevOps Architecture, DevOps with VSTS and Azure, VSTS Features, Competitors, Where can VSTS be Best Utilized?, Pros and Cons, Example of Reference Architecture, Why Azure? Pivotal Cloud Foundry, Key Competitors, Pros and Cons, Example of Reference Architecture, Concourse: CI-CD Tool by Pivotal, AWS, Reasons for Selecting AWS for DevOps, Features of AWS, AWS Tools and Services for Orchestrating DevOps Capability, Key Competitors, Pros and Cons. Example of Reference Architecture. Textbook: Chapter 6 (6.1.1 - 6.1.4), Chapter 9 (9.1 - 9.5)	
Module-5	9 Hours
DevOps Governance and Controls: What is DevOps Governance and Controls? How to Implement DevOps Governance? What are DevOps Governance Benefits? Implementing DevOps Controls. Adopting DevOps, What is a DevOps Playbook? Developing a Playbook Identify Target State from Current State, Preparing DevOps Playbook, Implementing DevOps Playbook, Playbook Implementation in New Application Development, Playbook Implementation for Existing Application. DevOps Emerging Trends: Strategy Trends , Independent Teams to Full Stack Developer Teams, DevOps to No-Ops, DevOps Resilience, Site Reliability Engineering (SRE) Technology Trends, AI and DevOps ,ChatBots. Docker Container. Textbook: Chapter 11 (11.1 - 11.4), Chapter 12 (12.1 - 12.3), Chapter 14 (14.1 - 14.2)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain DevOps concepts, culture, lifecycle and the role of DevOps tools in software development and operations.
CO2	Apply version control and build automation tools to manage source code, dependencies and software builds.
CO3	Implement Continuous Integration and Continuous Deployment (CI/CD) pipelines using DevOps tools to automate build, test and release processes.
CO4	Apply configuration management and infrastructure automation tools to deploy and manage applications across environments.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Deepak Gaikwad and Viral Thakkar, DevOps Tools from Practitioner's Viewpoint, 1st Edition, Wiley India Pvt. Ltd., 2019.
2. Gene Kim, Kevin Behr and George Spafford, The Phoenix Project: A Novel About

IT, DevOps and Helping Your Business Win, 5th Anniversary Edition, IT Revolution Press, 2018.

Reference Books:

1. Jez Humble and David Farley, Continuous Delivery: Reliable Software Releases through Build, Test and Deployment Automation, 1st Edition, Addison-Wesley Professional, 2010.
2. Gene Kim, Jez Humble, Patrick Debois and John Willis, The DevOps Handbook: How to Create World-Class Agility, Reliability and Security in Technology Organizations, 1st Edition, IT Revolution Press, 2016.

Web Links:

1. <https://www.geeksforgeeks.org/devops-tutorial/>
2. <https://www.javacodegeeks.com/2016/09/maven-vs-gradle-comparison.html>
3. <https://www.jenkins.io/doc/tutorials/>
4. <https://www.digitalocean.com/community/tutorials/an-introduction-to-ansible>
5. <https://learn.microsoft.com/en-us/azure/devops/pipelines/create-first-pipeline>

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	INTRODUCTION TO DATA STRUCTURES		
Course Code	BCS654A	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Introduce elementary data structures.
2. Analyze Linear Data Structures: Stack, Queues, Linked list
3. Analyze Non-Linear Data Structures: Trees, graphs
4. Assess appropriate data structure during program development/Problem Solving

Module-1	9 Hours
Introduction to Arrays and Pointers: Introduction, Linear Arrays, Representation of Linear Arrays in Memory, Traversing Linear Arrays, Inserting and Deleting, Pointers: Pointer Arrays. Strings: Introduction, Reading and Writing Strings, Operations on Strings, Arrays of Strings, Pointers and Strings. Textbook 2: Chapter 4 (4.1 - 4.5, 4.10) Textbook 1: Chapter 4 (4.1 - 4.4)	
Module-2	9 Hours
Structures and Unions: Introduction, Nested Structure, Array of Structure, Self-referential Structures, Unions. Stacks: Introduction, Array Representation of Stacks, Operations on Stack – Push, Pop and Peek Operation. Textbook 1: Chapter 5 (5.1 - 5.3, 5.5, 5.6), Chapter 7(7.1 - 7.3)	
Module-3	9 Hours
Queue: Introduction, Array Representation of Queues, Linked Representation of Queues, Types of Queue – Circular Queue and Deques. Linked List: Introduction, Singly Linked List, Circular Linked List. Textbook 1: Chapter 8 (8.1, 8.2, 8.4), Chapter 6 (6.1 - 6.3)	
Module-4	9 Hours
Linked List (Contd.): Doubly Linked List – Insert and Delete Operation.	

Trees: Introduction, Types of Trees, Create a Binary Tree from a General Tree, Traversing a Binary Tree, Applications of Trees.

Textbook 1: Chapter 6 (6.4), Chapter 9 (9.1 - 9.4, 9.6)

Module-5

9 Hours

Graph: Introduction, Graph Terminology, Representations of Graphs, Graph Traversal Algorithm, Applications of Graph.

Searching and Sorting: Introduction to Searching, Linear Search, Binary Search, Bubble Sort, Insertion Sort.

Textbook 1: Chapter 13 (13.1 - 13.2, 13.5, 13.6, 13.9), Chapter 14 (14.1 - 14.3, 14.7, 14.8)

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Implement fundamental data structures using arrays, pointers and structures.
CO2	Apply stacks and queues to evaluate expressions and solve linear data structure problems.
CO3	Implement memory-efficient programs using singly and circular linked lists.
CO4	Analyze tree and graph structures to model hierarchical and network relationships.

ASSESSMENT AND EVALUATION REGULATIONS OF OEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.

CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Reema Thareja, Data Structure using C, 3rd Edition, OXFORD University Press, 2023
2. Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.

Reference Books:

1. Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Ed, Universities Press, 2014.
2. E. Balagurusamy, C Programming and Data Structures, 4th Edition, McGraw-Hill, 2007.

Web links:

1. https://youtu.be/S47aSEqm_OI?si=34alu47wCWGdsxwN
2. <https://ds2-iiith.vlabs.ac.in/exp/selection-sort/index.html>
3. <https://ds1-iiith.vlabs.ac.in/exp/stacks-queues/index.html>
4. <https://ds1-iiith.vlabs.ac.in/exp/linked-list/basics/overview.html>
5. <https://ds1-iiith.vlabs.ac.in/List%20of%20experiments.html>
6. <https://ds1-iiith.vlabs.ac.in/exp/tree-traversal/index.html>
7. <https://ds1-iiith.vlabs.ac.in/exp/tree-traversal/depth-first%20traversal/dft-practice.html>
8. https://onlinecourses-archive.nptel.ac.in/noc18_cs25/preview

Self Study Topics:

1. Applications of Data Structures
 - 1.1 Organization of data in operating systems
 - 2.2 Organization of data in databases
 - 3.3 Organization of data in real-world problems

2. Sorting Techniques:

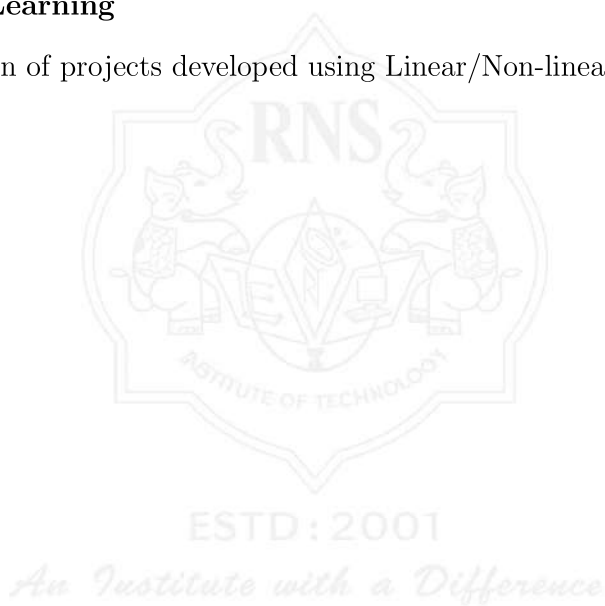
- 1.1 Selection sort
- 2.2 Merge Sort
- 3.3 Quick Sort

3. Advanced Tree Structure

- 1.1 AVL Trees, Red-Black Trees, B-Trees, B+ Trees

Activity Based Learning

- 1. Demonstration of projects developed using Linear/Non-linear data structures.



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	FUNDAMENTALS OF OPERATING SYSTEMS		
Course Code	BCY654B	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand the fundamental concepts and architecture of operating systems, including system structure, services and system calls.
2. Analyse and apply concepts of process, thread and CPU scheduling, along with inter-process communication and synchronization techniques.
3. Explain and evaluate memory management strategies, including paging, segmentation and virtual memory techniques.
4. Demonstrate knowledge of file systems, storage structures and protection mechanisms and use relevant system-level commands and APIs for managing OS resources.

Module-1

9 Hours

Introduction: What operating systems do, Computer System organization, Computer System architecture, Operating System operations, Resource Management.

Operating System Structures: Operating System Services, User - Operating System interface, System calls, System services, Operating system design and implementation, Operating System structure.

Textbook: Chapter 1 (1.1 - 1.5), Chapter 2 (2.1 - 2.4, 2.7, 2.8)

Module-2

9 Hours

Processes: Process concept, Process scheduling, Operations on processes, Inter process communication.

Threads & Concurrency: Overview, Multithreading models, Thread Libraries, Threading issues.

CPU Scheduling: Basic concepts, Scheduling Criteria, Scheduling Algorithms, Multiple-processor scheduling.

Textbook: Chapter 3 (3.1 - 3.4), Chapter 4 (4.1, 4.3 - 4.4, 4.6), Chapter 5 (5.1 - 5.3, 5.5)

Module-3

9 Hours

Synchronization Tools: Background, The critical section problem, Peterson's solution, Hardware Support for Synchronization, Mutex Locks, Semaphores.

Synchronization Examples: Classical problems of synchronization.

Deadlocks: System model, Deadlock characterization, Methods for handling deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, recovery from deadlock.

Textbook: Chapter 6 (6.1 - 6.6), Chapter 7 (7.1), Chapter 8 (8.1, 8.3 - 8.8)

Module-4

9 Hours

Main Memory: Background, Contiguous memory allocation, Paging, Structure of page table, Swapping.

Virtual Memory: Background, Demand paging, Copy-on-write, Page replacement, Allocation of frames, Thrashing.

Textbook: Chapter 9 (9.1 - 9.5), Chapter 10 (10.1 - 10.6)

Module-5

9 Hours

File System Interface: File concept, Access methods, Directory structure.

File System Implementation: File system structure, File system operations, Directory implementation, Allocation methods, Free space management.

Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix.

Textbook: Chapter 13 (13.1 - 13.3), Chapter 14 (14.1 - 14.5), Chapter 17 (17.1 - 17.2, 17.4 - 17.5)

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Interpret and explain the interactions among operating system components, including kernel functions, system calls and user interfaces, within typical system operations.
CO2	Apply CPU scheduling algorithms and process synchronization mechanisms to manage concurrent execution.
CO3	Analyse memory management techniques such as paging and segmentation to optimize memory utilization.
CO4	Evaluate file system structures and I/O management strategies to enhance overall system performance.
CO5	Simulate key operating system functionalities and system-level tasks using software tools and utilities.

ASSESSMENT AND EVALUATION REGULATIONS OF OEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Alternate Assessment (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE Evaluation is done for a maximum of 50 marks
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE are 20 out of allotted 50 marks to appear for SEE, with minimum 40% scored in every component of CIE.			

Textbooks:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Concepts, 9th Edition, Wiley-India, 2015.

Reference Books:

1. P.C.P. Bhatt, An Introduction to Operating Systems: Concepts and Practice, 4th Edition, PHI(EEE), 2014.
2. William Stallings, Operating Systems: Internals and Design Principles, 6th Edition, Pearson.
3. A.S. Tanenbaum, H. Bos, 2014, Modern Operating Systems, 4th Edition, Prentice-Hall of India, New Delhi.

Web links:

1. <https://pages.cs.wisc.edu/~remzi/OSTEP/>
2. <https://www.geeksforgeeks.org/operating-systems/>

3. https://www.tutorialspoint.com/operating_system/index.htm
4. <https://www.scaler.com/topics/operating-system/>
5. <https://www.youtube.com/watch?v=yK1uBHPdp30>
6. <https://www.youtube.com/watch?app=desktop&t=1s&v=d0iA2nNJpc0>

Self Learning Topics:

1. Virtual Machines: Study the concept of virtualization, including the architecture and functionality of virtual machines (VMs) and understand how VMs help in efficient resource utilization and system isolation.
2. Basic Linux Commands: Gain hands-on experience with essential Linux terminal commands used for file management, user permissions, process control and system monitoring.
3. Multithreading Demonstration using Programming Languages: Implement and compare multithreading concepts using:
 - a. Java – via Thread class and Runnable interface
 - b. Python – using the threading module
4. C Programming – using POSIX threads (pthreads): Overview of Different Operating Systems: Compare architecture, scheduling, memory management and file systems across these OS types: Linux, Windows, Unixandroid, MacOS, iOS, RTOS, Chrome OS, Solaris, Haiku OS, TinyOS

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	INTRODUCTION TO DBMS		
Course Code	BCI654C	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Provide a strong foundation in database concepts, technology and practice.
2. Practice SQL programming through a variety of database problems.
3. Understand the relational database design principles.
4. Apply normalization techniques to create well-structured relational database schemas.

Module-1	9 Hours
Introduction to Databases: Introduction, Characteristics of database approach, Actors on the Scene, Workers behind the Scene, Advantages of using the DBMS approach, Representing Complex Relationships among Data. Overview of Database Languages and Architectures: Data Models, Schemas and Instances. Three schema architecture and data independence, database languages and interfaces, The Database System environment, Centralized and Client/Server Architectures for DBMSs, Classification of Database Management Systems. Textbook: Chapter 1 (1.1 - 1.6.7), Chapter 2 (2.1 - 2.4)	
Module-2	9 Hours
Conceptual Data Modeling using Entities and Relationships: Entity Types, Entity Sets, Attributes and Keys Relationship Types, Relationship Sets, Roles and Structural Constraints, Weak Entity Types, Refining the ER Design for the COMPANY Database, ER Diagrams, Naming Conventions and Design Issues, Relationship Types of Degree Higher than Two, Another Example: A UNIVERSITY Database. Textbook: Chapter 3 (3.3 - 3.7, 3.9 - 3.10)	
Module-3	9 Hours
Mapping Conceptual Design into a Logical Design: Relational Database Design using ER-to-Relational mapping. Relational Model: Relational Model Concepts, Relational Model Constraints and relational database schemas, Update operations, transactions and dealing with constraint violations. Textbook: Chapter 9 (9.1 - 9.2), Chapter 5 (5.1 - 5.3)	

Module-4	9 Hours
Relational Algebra: Unary Relational Operations: SELECT and PROJECT, Relational Algebra Operations from Set Theory, Binary Relational Operations: JOIN and DIVISION. additional relational operations (aggregate, grouping, etc.) Examples of Queries in relational algebra. SQL: SQL data definition and data types, Schema change statements in SQL, specifying constraints in SQL, retrieval queries in SQL, INSERT, DELETE and UPDATE statements in SQL, Additional features of SQL Textbook: Chapter 9 (8.1 - 8.5), Chapter 6 (6.1 - 6.5)	
Module-5	9 Hours
Basics of Functional Dependencies and Normalization for Relational Databases: Informal design guidelines for relation schema, Functional Dependencies, Normal Forms Based on Primary Keys, General Definitions of Second and Third Normal Forms, Boyce-Codd Normal Form, Multivalued Dependency and Fourth Normal Form, Join Dependencies and Fifth Normal Form. Textbook: Chapter 14 (14.1 - 14.7)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Understand the fundamental concepts, architecture and components of Database Management Systems (DBMS).
CO2	Understand real-world problem scenarios by designing Entity-Relationship (ER) models.
CO3	Apply ER-to-relational mapping techniques to transform conceptual designs into logical database schemas. Analyze and use SQL queries for efficient database definition, creation and data manipulation.
CO4	Apply relational algebra operations to formulate SQL statements for DDL and DML.
CO5	Evaluate relational schemas using functional dependencies and apply normalization to eliminate anomalies.

ASSESSMENT AND EVALUATION REGULATIONS OF OEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.

Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE Evaluation is done for a maximum of 50 marks
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbook:

1. R. Elmasri and S. B. Navathe, Fundamentals of Database Systems, 7th Edition. Boston, MA: Pearson, 2017.

Reference Books:

1. R. Ramakrishnan and J. Gehrke, Database Management Systems, 3rd Edition. New York, NY: McGraw-Hill, 2014.
2. C. J. Date, A. Kannan and S. Swamynathan, An Introduction to Database Systems, 8th Edition. Noida, India: Pearson, 2006.

Self Study Topics:

1. Advanced Queries: Specifying constraints as assertions and action triggers, Views.
2. NOSQL Databases, Document-Based NOSQL Systems and MongoDB
3. Column-Based NOSQL Systems and Graph Databases.
4. Transaction Processing, Concurrency Control and Recovery

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	FOUNDATIONS OF DATA SCIENCE		
Course Code	BCD654D	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand the fundamental concepts of data science.
2. Explain statistical and probabilistic concepts for data-driven decision making.
3. Apply the data science methodology and analytics life cycle to solve real-world problems.
4. Analyse and implement core machine learning techniques.
5. Explore and apply text mining and basic natural language processing techniques.

Module-1	9 Hours
Importance of Data Science: Importance of Data Science, Need for Data Science, What Is Data Science, Data Science Process, Business Intelligence and Data Science, Prerequisites for a Data Scientist, Components of Data Science, Tools and Skills Needed. Data Science Tool: Basics of Python for Data Science, Python Libraries: DataFrame Manipulation with pandas and NumPy Textbook: Chapter 1 (1.1 - 1.8), Chapter 7 (7.1 - 7.2)	
Module-2	9 Hours
Statistics and Probability: Data Types, Variable Types, Statistics, Sampling Techniques and Probability, Information Gain and Entropy, Probability Theory, Probability Types, Probability Distribution Functions, Bayes' Theorem, Inferential Statistics. Textbook: Chapter 2 (2.1 - 2.11)	
Module-3	9 Hours
Data Science Methodology: Analytics for Data Science, Examples of Data Analytics, Data Analytics Life Cycle: Data Discovery, Data Preparation, Model Planning, Model Building, Communicate Results, operationalization. Textbook: Chapter 4 (4.1 - 4.4)	

Module-4	9 Hours
Data Science Methods and Machine Learning: Regression Analysis, Linear Regression, Logistic regression, Multimodal logistic regression, Time-series model, Machine Learning: Decision trees, naive bayes, support vector, machines, nearest neighbour learning, clustering, Confusion matrix Textbook: Chapter 5 (5.1 - 5.3)	
Module-5	9 Hours
Data Analysis and Text Mining: Text Analytics: Text analytics subtasks, Cleaning and parsing, searching and retrieval, Text mining, Part-of-speech tagging, Stemming, Lemmatization, Basic text analysis steps, Introduction to Natural Language Processing. Textbook: Chapter 6 (6.1 - 6.4)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Understand the fundamental concepts of Data Science, including its importance, process, components and the role of SQL in data analysis and decision-making.
CO2	Apply statistical methods, probability distributions and Bayes' Theorem to analyze data types and calculate metrics like Entropy and Information Gain.
CO3	Execute the Data Analytics Life Cycle, from initial data discovery and preparation to model building and the operationalization of results.
CO4	Analyze model performance using evaluation metrics such as the confusion matrix and interpret the results to assess predictive accuracy.
CO5	Demonstrate the ability to use basic Natural Language Processing (NLP) techniques and tools for real-world text analysis applications.

ASSESSMENT AND EVALUATION REGULATIONS OF OEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.

Total CIE (A+B)	50	20	CIE Evaluation is done for a maximum of 50 marks
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Sanjeev J. Wagh, Manisha S. Bhende and Anuradha D. Thakare, Fundamentals of Data Science, 1st Edition, CRC press, 2021.

Reference Books:

1. Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, An Introduction to Statistical Learning: with Applications in R, Springer, 1st Edition, 2013
2. Peter Bruce and Andrew Bruce, Practical Statistics for Data Scientists, 2017, O'Reilly
3. Cathy O'Neil, Rachel Schutt, Doing Data Science - Straight Talk from the Frontline, O'Reilly Media, 2013.

Web links:

1. <https://www.udemy.com/course/data-science-for-the-beginner/?couponCode=PMNVD2025>
2. <https://www.oreilly.com/library/view/doing-data-science/9781449363871/toc01.html>
3. <https://docs.python.org/3/tutorial/>
4. <https://nptel.ac.in/courses/106106212>

Self Study Topics:

1. Difference between Data Science vs Business Intelligence vs AI
2. Central Limit Theorem (intuitive understanding)
3. CRISP-DM vs Data Analytics Life Cycle
4. Comparison of ML algorithms and analyze datasets.

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING LABORATORY		
Course Code	BCSL606	CIE MARKS	50
(L : T : P)	(0:0:2)	SEE MARKS	50
Credits	1	EXAM HOURS	3

COURSE LEARNING OBJECTIVES:

The purpose of the course is to facilitate the learner to:

1. Understand Artificial Intelligence concepts such as intelligent agents and search techniques.
2. Implement search algorithms for solving real-world problems.
3. Evaluate models using probabilistic approaches and neural networks.
4. Demonstrate intelligent systems using decision trees and reinforcement learning methods.

Prerequisites:

1. Basic knowledge of Python programming.
2. Understanding of data structures (arrays, lists, graphs, trees).
3. Fundamental knowledge of mathematics and statistics (probability, linear algebra basics).
4. Familiarity with basic problem-solving and algorithm design concepts.

Program 1 - Agent Simulation in Observable and Stochastic Environments

Consider a robot operating in a smart home environment where some rooms are fully observable while others have limited visibility.

- (a) Design and implement a Python program to simulate the agent's behavior in both fully observable and partially observable environments.
- (b) Extend the scenario to include uncertainty in actions (e.g., movement failure or random obstacles) and implement the agent in deterministic and stochastic environments. Analyze and compare the agent's performance in each case.

Program 2 - Problem-Solving Agent for City Navigation

Consider a robot navigating a city map to reach a destination from a starting point.

- (a) Develop a Python program to model this as a problem-solving agent by defining the initial state, goal state and possible actions.
- (b) Extend the program to apply a suitable search algorithm to find the optimal path and display the sequence of steps taken by the agent to reach the destination.

Program 3 - Graph Traversal Using BFS and DFS in a Social Network

Consider a social network where users are connected as a graph.

- (a) Develop a Python program to use Breadth First Search (BFS) to find and display all reachable users starting from a given user node.
- (b) For the same network, implement Depth First Search (DFS) and compare its traversal with BFS in terms of order of exploration, memory usage and overall execution behavior.

Program 4 - Optimal Route Finding Using UCS and A* Search

A delivery robot must find the lowest-cost route between locations in a city represented as a weighted graph.

- (a) Develop a Python program using Uniform Cost Search (UCS) to determine the optimal path and total travel cost from a source to a destination.
- (b) The robot is further enhanced with heuristic knowledge (e.g., straight-line distance to destination). Implement the A* search algorithm and compare its performance with UCS in terms of path cost, number of nodes explored and efficiency.

Program 5 - Optimal Game Playing Using Minimax and Alpha-Beta Pruning

Consider a scenario where you are developing an AI opponent for a Tic-Tac-Toe game that always plays optimally against a human player.

- (a) Write a Python program using the Minimax algorithm to enable the system to choose the best possible move at each step.
- (b) As the game complexity increases, optimize the decision-making process by incorporating Alpha-Beta pruning. Modify the program to reduce unnecessary computations and compare its efficiency with the basic Minimax approach in terms of execution time and nodes evaluated.

Program 6 - Data Preprocessing and Model Evaluation for Student Performance Data

A dataset collected from a student performance system contains missing values, inconsistent formats and features with different scales.

- (a) Develop a Python program to preprocess the dataset by handling missing values and applying normalization techniques.
- (b) Split the dataset into training and testing sets, build a simple model and evaluate its performance using appropriate metrics.

Program 7 - Customer Classification Using KNN and Weighted KNN

A retail company wants to classify customers into different categories based on their purchase behavior.

- (a) Develop a Python program using the K-Nearest Neighbors (KNN) algorithm to classify customers based on features such as age, income and spending score.
- (b) Enhance the program by implementing Weighted KNN to improve classification accuracy and compare its performance with standard KNN for the given dataset.

Program 8 - House Price Prediction Using Simple and Multiple Linear Regression

A company wants to predict house prices based on different factors such as area, number of bedrooms and location features. Initially, they plan to use a single feature (area) and later extend the model using multiple features for better accuracy.

- (a) Develop a Python program to implement Simple Linear Regression using one feature (e.g., area) to predict house prices.
- (b) Extend the program to implement Multiple Linear Regression using multiple features and analyze the improvement in prediction accuracy.

Program 9 - Customer Churn Prediction Using Logistic Regression

A telecom company wants to predict whether a customer will churn based on features such as usage, billing and tenure.

- (a) Develop a Logistic Regression model to classify customers as churn or not.
- (b) Evaluate the model's performance using accuracy, confusion matrix, precision and recall and interpret how well it identifies potential churn customers.

Program 10 - Spam Email Classification Using Naive Bayes

A company wants to automatically classify emails as spam or not spam based on features such as keywords and frequency.

- Develop a Python program using Naïve Bayes classifier to perform this classification task.
- Extend the program to handle continuous features such as email length and word frequency using Gaussian Naïve Bayes and analyze the improvement in classification performance.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Implement intelligent agents for real-world settings.
CO2	Apply search algorithms for finding efficient solutions.
CO3	Implement machine learning models for classification and regression tasks.
CO4	Apply Bayesian learning and neural networks for solving real-world problems.
CO5	Demonstrate applications using decision trees and reinforcement learning techniques.

ASSESSMENT AND EVALUATION REGULATIONS OF PCCL

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE Practical (A)	30	-	Marks awarded for Weekly Conduction of Experiments and Submission of Laboratory records, scaled to 30 marks.
CIE Practical Test (B)	20	-	Average of marks of Two tests, each conducted for 100 marks covering all experiments, scaled to 20.
Total CIE Practical (A+B)	50	20	Sum of Marks of Experiments, Record and Test shall be CIE marks awarded for a maximum of 50.
SEE (C)	50	20	SEE is a lab exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.

CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Self Study Topics:

1. Fundamentals of Artificial Intelligence and Intelligent Agents.
2. Search Algorithms: BFS, DFS, UCS and A* Search Techniques.
3. Machine Learning Concepts: Supervised vs Unsupervised Learning.
4. Classification Techniques: KNN, Naïve Bayes, Logistic Regression.
5. Neural Networks and Basics of Deep Learning.

Activity Based Topics:

1. Implementation of Search Algorithms on real-world problems (Path Finding/Game Playing).
2. Machine Learning Model Building and Evaluation using real datasets.
3. Mini Project on AI/ML application (e.g., prediction or classification problem).

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	AUTOMATED SOFTWARE TESTING		
Course Code	BCSL657A	CIE MARKS	50
(L : T : P)	(0:0:2)	SEE MARKS	50
Credits	1	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Introduce the fundamental concepts and importance of software testing and test automation.
2. Provide knowledge of automated testing tools and techniques used in the software development lifecycle.
3. Develop the ability to design and execute automated test cases for software applications.
4. Familiarize students with test automation frameworks, test data management and reporting mechanisms.
5. Expose learners to Continuous Integration and Continuous Deployment (CI/CD) practices used in automated testing environments.

Prerequisites:

1. Installation and setup of Katalon Studio; web locators, selectors hub and browser developer tools.
2. Installation and configuration of Selenium RC and Selenium WebDriver.

Program 1 - Test Automation Using Katalon Studio

A software testing team is assigned to automate the testing process of a web application. Using Katalon Studio, the team must design and execute automated test cases by exploring different modes such as Record Mode, Manual Mode and Script Mode. During the automation process, the testers should utilize built-in keywords to perform actions such as navigation, input validation and verification. The objective is to effectively implement test automation and ensure that the application functions correctly under various test conditions.

Program 2 - Cross-Browser Testing Using Katalon Recorder

A software testing team is assigned to validate the functionality of a web application across different browsers. The testers use the Katalon Recorder tool in Chrome and Firefox to record user interactions, replay the automated test cases and validate the

application's functionality. Through this process, the team ensures that the application behaves consistently across multiple browsers and that the recorded test cases execute successfully.

Program 3 - Web Element Identification Using Web Spy and Object Repository

A software testing team is developing automated test scripts for a web application. During the automation process, the testers need to identify and manage web elements using Web Spy and the Object Repository in the automation tool. By applying appropriate locator strategies

such as ID, name, XPath and CSS selectors, the team ensures that the elements are accurately identified. This approach helps create reliable and maintainable automated test scripts for effective test execution.

Program 4 - Web UI Automation Using Predefined Keywords

A software testing team is assigned to automate the testing of a web application using an automation tool. The testers must utilize predefined Web UI keywords to perform various operations such as browser and window handling, interacting with form controls (checkboxes, textboxes and file uploads), capturing screenshots, handling alerts and performing mouse actions like hover and drag-and-drop. Additionally, the testers must implement keystroke operations to simulate user inputs. The objective is to ensure that the automated scripts accurately replicate real user interactions and verify the correct functioning of the web application.

Program 5 - Converting Manual Test Cases to Selenium Automation Scripts

A software testing team is responsible for validating the functionality of a web application. Initially, the testers design manual test cases based on the application requirements. Later, the team converts these manual test cases into Selenium automation scripts to automate the testing process. The objective is to ensure that the automated scripts effectively validate the application's functionality and improve the efficiency and reliability of the testing process.

Program 6 - Test Suite Creation and Execution Using Selenium IDE

A software testing team is assigned to automate the testing of a web application using Selenium IDE. The testers must create and organize a test suite that includes at least four different test cases covering key functionalities of the application. The team records or scripts the test steps in Selenium IDE and executes the test suite to verify whether the application behaves as expected during automated testing.

Program 7 - Automated Test Execution Across Multiple Web Applications

A software testing team is responsible for validating the functionality of several web applications hosted on different websites. The testers must configure and execute automated test suites to verify key functional workflows across these heterogeneous web environments. During the testing process, the team ensures that the test execution is consistent and the results are properly analyzed and reported, helping identify any defects or inconsistencies in the application behavior across different web platforms.

Program 8 - Automating Login Functionality Using Selenium

A software testing team is assigned to automate the login functionality of a web application using Selenium. The testers must develop a moderately complex Selenium test script that validates input fields such as username and password, automates the authentication process and verifies both successful and unsuccessful login attempts. The objective is to ensure that the login workflow functions correctly and that the application properly handles valid and invalid user credentials.

Program 9 - Identifying and Counting Web Objects Using Locator Strategies

A software testing team is tasked with analyzing the structure of a web application page as part of automated testing. Using Selenium, the testers must develop a test script that identifies web objects such as links, buttons, text fields and images by applying appropriate locator strategies (e.g., ID, name, XPath, or CSS selectors). The script should then compute the total number of these web elements present on the webpage and validate the results to ensure that all required objects are correctly detected during the automation process.

Program 10 - Designing Data-Driven, Keyword-Driven and Hybrid Automation Frameworks

A software testing team is working on improving the efficiency and scalability of their automated testing process for a large web application. To achieve this, the team must design and implement advanced test automation frameworks such as data-driven, keyword-driven and hybrid frameworks using appropriate automation tools. These frameworks should support parameterized testing, enable the use of reusable test components and provide a scalable and maintainable automation solution for executing multiple test cases efficiently across different test scenarios.

Program 11 - Test Data Management Using Excel and CSV File Integration

A software testing team is tasked with improving the flexibility of their automated testing process. To achieve this, the testers must implement test data management by integrating external data sources such as Excel and CSV files into their automation framework. By applying parameterization techniques, the team should enable the execution of test cases with multiple sets of input data, ensuring flexible, reusable and data-driven test execution for validating various application scenarios.

Program 12 - CI/CD Integration of Automated Testing Using Jenkins

A software testing team aims to integrate automated testing into the software development lifecycle using CI/CD practices. As part of a demonstration, the testers must configure Jenkins to trigger automated test execution whenever a new build is generated. The process should illustrate automated build triggering, integration of test scripts with the CI pipeline and basic result reporting, helping the team understand how automated tests can be executed continuously within a CI/CD environment.

Program 13 - Test Execution Reporting, Logging and Failure Analysis

A software testing team is assigned to evaluate the quality of a newly developed web application. During automated test execution, the team needs to generate and analyze test execution reports and logs to monitor the testing process. In case of test failures, the system should automatically capture screenshots to help identify the cause of the issue. Based on the generated reports and captured evidence, the testers must interpret the results to assess the application quality and identify defects that need to be fixed before release.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Understand the architecture, key features and components of automation tools such as Katalon Studio and Selenium.
CO2	Design the functional and regression test scenarios for web applications using test automation tools.
CO3	Develop reusable test cases, automation scripts and maintain object repositories for efficient test execution.
CO4	Execute automated test suites on web applications across multiple browsers and different testing environments to ensure cross-platform compatibility.
CO5	Apply basic Continuous Integration and Continuous Deployment (CI/CD) concepts to run automated tests and analyze test reports.

ASSESSMENT AND EVALUATION REGULATIONS OF AEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE Practical (A)	30	-	Marks awarded for Weekly Conduction of Experiments and Submission of Laboratory records, scaled to 20 marks.
CIE Practical Test	20	-	Average of marks of Two tests, each conducted for 100 marks covering all experiments, scaled to 20.

Total CIE Practical (A+B)	50	20	Sum of Marks of Experiments, Record and Test shall be CIE marks awarded for a maximum of 50.
SEE (C)	50	20	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE+SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Elfriede Dustin, Jeff Rashka and John Paul, Automated Software Testing: Introduction, Management and Performance, 1st Edition, Addison-Wesley Professional, 1999.

Reference Books:

1. Mark Fewster and Dorothy Graham, Software Test Automation: Effective Use of Test Execution Tools, 1st Edition, Addison-Wesley (ACM Press), 1999.
2. Unmesh Gundecha, Selenium Testing Tools Cookbook, 1st Edition, Packt Publishing, 2012.

Web Links:

1. <https://www.parasoft.com/learning-center/automated-testing-guide>

Self Study Topics:

1. Evolution of software testing and the need for automation.
2. Comparison of manual testing vs automated testing.
3. Types of test automation: Unit, Integration, System, Regression, Smoke, Sanity.
4. Test automation frameworks: Linear, Modular, Data-Driven, Keyword-Driven, Hybrid.
5. Overview of popular automation tools (Selenium, JUnit, TestNG, Cypress – conceptual).

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	ETHICAL HACKING AND VULNERABILITY ASSESSMENT		
Course Code	BCSL657B	CIE MARKS	50
(L : T : P)	(0:0:2)	SEE MARKS	50
Credits	1	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Learn core concepts and techniques of network, web and application penetration testing.
2. Identify and exploit common vulnerabilities such as XSS, SQLi, Command Injection, File Upload, Session Attacks and XXE.
3. Use industry-standard tools (Nmap, Wireshark, Burp Suite, Hashcat, etc.) for reconnaissance and exploitation.
4. Understand security risks and apply mitigation through secure coding and defensive practices.

Program 1 - Network Scanning using Nmap/Zenmap

A university campus network administrator has noticed irregular traffic patterns and suspects that unknown devices and unauthorized services may be active within the internal network. As a student intern in the cyber security team, you are assigned the task of conducting a controlled network reconnaissance exercise. Your responsibility is to ethically scan the assigned lab network to identify active hosts, discover open ports, detect running services and estimate operating systems in use.

Program 2 - Packet Capture & Traffic Analysis using Wireshark

A network security team is investigating unusual behaviour within a controlled lab network after detecting abnormal traffic patterns. As part of the incident analysis, live packets are captured and examined to study protocol communication and identify suspicious activity. This scenario simulates real-world traffic monitoring and threat detection using Wireshark.

Program 3 - Password Cracking

A cybersecurity audit team is conducting an evaluation of authentication mechanisms within a controlled lab system. As part of the assessment, sample password hashes are tested through an ethical password audit to demonstrate common weaknesses in password selection. This scenario simulates real-world security testing and highlights the importance of enforcing strong password policies using tools such as Hashcat and John the Ripper.

Program 4 - Cross-Site Scripting (XSS)

A web security assessment team is assigned to examine a deliberately vulnerable lab application to evaluate client-side security weaknesses. The team performs controlled testing to identify and exploit Cross-Site Scripting (XSS) flaws and study how improper input handling leads to attacks. This scenario simulates a real-world security audit and emphasizes practical mitigation techniques.

Program 5 - File Upload Vulnerability

A security audit team is evaluating a web application that allows users to upload files, after reports of suspicious server behavior. The team conducts controlled testing to demonstrate how weak upload validation can be bypassed to introduce malicious files. This scenario simulates real-world exploitation of insecure file upload mechanisms and highlights the need for secure validation practices using DVWA and Burp Suite.

Program 6 - Command Injection

A security assessment team is reviewing a custom web application after signs of abnormal server-side command execution are observed. During controlled testing, the application is examined to demonstrate how improper input handling can allow command injection attacks. This scenario simulates a real-world vulnerability assessment and emphasizes the importance of input validation and secure coding practices using DVWA and Burp Suite.

Program 7 - Man-in-the-Middle (MITM) Attack

A network security team is investigating a suspected breach where communication between two systems may have been intercepted. In a controlled lab setup, traffic is monitored and manipulated to demonstrate how a Man-in-the-Middle (MITM) attack can occur. This scenario simulates real-world interception techniques and highlights the risks of unsecured network communication using Ettercap and Wireshark.

Program 8 - Session Hijacking and Fixation

A web application security team is analysing a vulnerable login system after reports of unauthorized account access. In a controlled lab environment, session management mechanisms are tested to demonstrate how session hijacking and session fixation attacks can occur. This scenario simulates real-world exploitation of weak session handling and highlights the importance of secure session management practices using DVWA, Burp Suite and browser developer tools.

Program 9 - SQL Injection using Burp Suite

A database-driven web application is suspected of exposing sensitive information due to improper input handling. A security assessment team conducts controlled testing to identify and exploit SQL injection vulnerabilities through intercepted requests and payload manipulation. This scenario simulates a real-world web application audit and demonstrates database attack techniques using Burp Suite and DVWA.

Program 10 - XML External Entity (XXE) Injection

A development team is reviewing an XML-based web service after abnormal data access behavior is detected. During a controlled security assessment, the application is tested to demonstrate how improperly configured XML parsers can be exploited through XML External Entity (XXE) injection. This scenario simulates a real-world vulnerability investigation and highlights the risks of insecure XML processing using Burp Suite and a vulnerable XML application.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Apply network scanning and traffic analysis techniques.
CO2	Identify common web and system vulnerabilities.
CO3	Analyze password and injection attacks to assess system vulnerabilities.
CO4	Apply secure coding practices to mitigate software vulnerabilities.

ASSESSMENT AND EVALUATION REGULATIONS OF AEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE Practical (A)	30	-	Marks awarded for weekly conduction of experiments and submission of laboratory records, scaled to 30 marks.
CIE Practical Test (B)	20	-	Average of marks of two tests, each conducted for 100 marks covering all experiments, scaled to 20.
Total CIE Practical (A+B)	50	20	Sum of marks of experiments, record and test shall be CIE marks awarded for a maximum of 50.
SEE (C)	50	20	SEE is a lab exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, Band C.
The minimum marks to be secured in CIE are 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Dafydd Stuttard and Marcus Pinto, The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws, 2nd Edition, John Wiley & Sons, Inc., 2011. <https://portswigger.net/wahh>
2. Gus Khawaja, Practical Web Penetration Testing: Secure your web applications against modern attack vectors, 1st Edition, Packt Publishing, 2018. <https://slink.com/zw6vwbc>
3. OWASP, Web Security Testing Guide (WSTG), Version 4.2, Open Web Application Security Project (OWASP) Foundation, 2020. <https://owasp.org/www-project-web-security-testing-guide/latest/>

Official Tool Documentation:

1. <https://portswigger.net/burp/documentation>

High-Quality Tutorials & Training:

1. <https://portswigger.net/web-security>
2. <https://cheatsheetseries.owasp.org/>
3. <https://slink.com/w4h54hb>

Trusted YouTube Education Channels:

1. <https://www.youtube.com/@HackerSploit>
2. <https://www.youtube.com/@davidbombal>
3. https://www.youtube.com/@_CryptoCat

Reliable Academic Reference:

1. <https://www.nist.gov/cyberframework>

Self-Learning Topics:

1. Test automation frameworks: Linear, Modular, Data-Driven, Keyword-Driven, Hybrid
2. Overview of popular automation tools (Selenium, JUnit, TestNG, Cypress – conceptual)

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	MLOPS		
Course Code	BCSL657C	CIE MARKS	50
(L : T : P)	(0:0:2)	SEE MARKS	50
Credits	1	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand the fundamentals of MLOps and its role in managing the machine learning lifecycle.
2. Gain practical experience in using modern tools such as Git, DVC, MLflow, Scikit-learn.
3. Develop skills in data pre-processing, model training, deployment and monitoring.
4. Apply ML workflow and MLOps practices for problem-solving and apply MLOps techniques to real-world datasets.

Program 1 - Environment Setup Verification for Machine Learning using Python and Library Version Checking

A student is working on a Machine Learning project using Python and libraries like NumPy, Pandas and Scikit-learn. The model runs successfully on their system. However, when the same project is shared with a teammate, it fails due to differences in Python and library versions. To avoid such issues and ensure a reproducible ML environment, the student decides to write a program that checks and documents the versions of Python and required ML libraries.

Tools / Libraries: python, Numpy, Pandas, Scikit-learn

Program 2 - Exploratory Data Analysis (EDA): Dataset Loading and Basic Inspection using Pandas

A data science student is working on a Machine Learning project and receives a dataset in CSV format. Before building any model, the student needs to understand the dataset—its structure, size and data quality. To achieve this, the student writes a Python program to load the dataset and perform basic inspection such as viewing rows, checking data types, identifying missing values and understanding overall dataset information.

Tools / Libraries: Python, pandas, CSV dataset

Program 3 - Data Preprocessing: Detection and Handling of Missing Values in Machine Learning Dataset

A student is developing a Machine Learning model using a dataset. Before training the model, the student wants to ensure the dataset is clean and does not contain missing (null) values, as these can affect model accuracy and cause errors. To ensure data quality, the student writes a Python program to validate the dataset by checking for missing values and handling them appropriately.

Tools / Libraries : Python, pandas, CSV dataset

Program 4 - Feature Scaling and Normalization of Dataset using Scikit-learn

A student is building a Machine Learning model using a dataset that contains features with different scales—for example, age (0–100) and income (in lakhs). Because these values vary widely, features with larger scales can dominate the model and reduce accuracy. To solve this, the student applies feature normalization so that all features are brought to a similar scale before training the model.

Tools / Libraries : Python, Pandas, Scikit-learn

Program 5 - Dataset Splitting into Training and Testing Sets for Model Evaluation

A student has prepared a clean and normalized dataset for a Machine Learning model. If the model is trained and tested on the same data, it may show very high accuracy but fail on new, unseen data (over fitting).

To ensure an unbiased evaluation, the student splits the dataset into:

Training set → used to train the model

Testing set → used to evaluate performance

Tools / Libraries : Python, Pandas, Scikit-learn

Program 6 - Machine Learning Experiment Tracking using MLflow

While developing a Machine Learning model, different algorithms and parameter settings are tested. Manually recording results is difficult and error-prone. To overcome this, MLflow is used to automatically log experiment details and compare model performance efficiently.

Tools / Libraries : Python, MLflow, Scikit-learn, Pandas, Numpy

Program 7 - Model Persistence: Saving and Loading Trained Machine Learning Models using Joblib

A student trains a Machine Learning model to predict student results. Training the model every time is time-consuming. To make the system efficient, the student saves the trained model to a file. Later, the saved model is loaded and used directly for predictions.

Tools / Libraries : Python, Pandas, Scikit-learn, Joblib

Program 8 - Data Drift Detection using Statistical Analysis for Monitoring Model Performance

A student trains a Machine Learning model using historical student data. After deployment, new data starts coming in. Over time, the characteristics of the new data may change (for example, marks distribution or income levels). If the new data is significantly different from the training data, the model's performance may degrade. To prevent this, the student performs data drift detection by comparing statistical properties of the training and new datasets.

Tools / Libraries : Python, Pandas, Numpy, Scipy

Program 9 - Automated Machine Learning Pipeline for Preprocessing and Classification using Scikit-learn

A student is working on a classification problem using a dataset that contains raw, unprocessed data. The dataset requires multiple preprocessing steps such as handling missing values, scaling features and then applying a classification algorithm. Instead of performing these steps manually each time, the student creates an ML pipeline to automate preprocessing and model training in a structured and reusable way

Tools / Libraries : Python, Scikit-learn

Program 10 - Model Performance Monitoring and Accuracy Evaluation on Streaming Data

A student deploys a Machine Learning model to predict student results. Initially, the model performs well. However, as new data arrives over time, the model's accuracy may decrease due to changes in data patterns. To ensure reliability, the student continuously monitors the model's performance on new data and checks whether accuracy drops below an acceptable threshold.

Tools / Libraries : Python, Scikit-learn, Pandas

Program 11 - Reusable Machine Learning Model Inference using Saved Models with Joblib

A student has already trained and saved a Machine Learning model to predict student results. Instead of retraining the model every time, the student wants to reuse the saved model to make predictions on multiple new inputs (e.g., new students' data). This improves efficiency and allows quick predictions in real-world applications.

Tools / Libraries : Python, Scikit-learn, Pandas, Joblib

Program 12 - Model Explainability and Fairness Analysis using SHAP and LIME for Classification Models

A student develops a Machine Learning model to predict student results. After deployment, it becomes important to understand why the model is making certain predictions and ensure that it is not biased toward any group (e.g., gender, category).

To achieve this, the student uses:

Model Explain ability → to understand feature importance

Bias Detection → to check fairness across groups

Tools / Libraries : Python, SHAP, LIME, scikit-learn

Train a classification model (Iris / Credit dataset), generate SHAP or LIME explanations, analyse feature importance, detect bias in predictions across classes

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Configure the ML development environment and document it for reproducibility.
CO2	Implement dataset versioning and experiment tracking using DVC and MLflow.
CO3	Design data preprocessing and feature engineering pipelines for machine learning applications.
CO4	Build and evaluate machine learning models using Scikit-learn.
CO5	Deploy models, detect data drift and interpret model predictions.

ASSESSMENT AND EVALUATION REGULATIONS OF AEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE Practical (A)	30	-	Marks awarded for Weekly Conduction of Experiments and Submission of Laboratory records, scaled to 30 marks.
CIE Practical Test (B)	20	-	Average of marks of Two tests, each conducted for 100 marks covering all experiments, scaled to 20.
Total CIE (A+B)	50	20	Sum of Marks of Experiments, Record and Test shall be CIE marks awarded for a maximum of 50.
SEE (C)	50	20	SEE is a lab exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE+SEE (A+B+C)	100	40	Final marks calculation.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Noah Gift and Alfredo Deza, Practical MLOps, O'Reilly, 1st Edition, 2011.
2. Mark Treveil, Nicolas Omont, et al., Introducing MLOps, O'Reilly, 1st Edition, 2011.

Reference Books:

1. Ben Wilson, Machine Learning Engineering in Action, Manning Publications, 1st Edition, 2022.
2. Chip Huyen, Designing Machine Learning Systems: An Iterative Process for Production-Ready Applications, O'Reilly, 1st Edition, 2022.

Self Learning Components:

1. Learned to use MLflow for tracking experiments, parameters and model performance.
2. Gained understanding of data preprocessing techniques such as handling missing values and feature scaling.
3. Explored model deployment basics using saved models for real-time predictions.
4. Understood the importance of model monitoring to detect data drift and performance degradation.
5. Studied model explainability and bias detection to ensure reliable and fair predictions.

Reference Links:

1. <https://slink.com/vmoinzm>
2. <https://mlops-guide.github.io/Workflow/>
3. <https://l1nq.com/g12qdy1>

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	SYSTEM DESIGN AND SCALABLE ARCHITECTURE		
Course Code	BCSL657D	CIE MARKS	50
(L : T : P)	(0:0:2)	SEE MARKS	50
Credits	1	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand the fundamentals of system design and building scalable architectures.
2. Understand and implement commonly used cache replacement policies in large scale systems.
3. Understand and implement common load balancing schemes used in large scale systems.
4. Understand the usage of and implement programs for common application layer protocols.
5. Understand the concept of scaling of databases in large scale systems and implement sharding and replication of MongoDB databases.

Prerequisites:

1. Basic knowledge of programming in Java.
2. Basic knowledge of working on a Linux-based system.
3. Basic knowledge of common application layer protocols.
4. Basic knowledge of NoSQL databases.

Program 1 - FIFO Cache Management Using LinkedList

Scenario: Consider a list of n integer valued items and a cache of capacity k , where $k \leq n$. Develop a program in Java to implement a First In First Out (FIFO) cache using a LinkedList. The value of n is 10, with the values of inputs varying randomly from 1 to 5 and the cache size k is 3. The program should output the status of cache after each input item is accessed.

Program 2 - LRU Cache Implementation Using Queue and HashMap

Scenario: Consider a list of n integer valued items and a cache of capacity k , where $k \leq n$. Develop a program in Java to implement a Least Recently Used (LRU) cache using Queue and HashMap. The value of n is 10, with the values of inputs varying randomly from 1 to 5 and the cache size k is 3. The program should output the status of cache after each input item is accessed.

Program 3 - LFU Cache Management Using HashMap and LinkedHashMap

Scenario: Consider a list of n integer valued items and a cache of capacity k , where $k < n$. Develop a program in Java to implement a Least Frequently Used (LFU) cache using HashMap and LinkedHashMap. The value of n is 10, with the values of inputs varying randomly from 1 to 5 and the cache size k is 3. The program should output the status of cache after each input item is accessed.

Program 4 - Round Robin Task Assignment to Servers

Scenario: Consider an input of 10 tasks which are to be assigned to three servers. Develop a program in Java to implement a Round Robin load balancer for assigning the above tasks to servers. Consider each task to be represented by a unique integer. Display the workload on each server following the addition of each task.

Program 5 - Server Load Distribution Using Least Connections Algorithm

Scenario: Consider an input of 10 tasks which are to be assigned to three servers. Assign a random initial workload of one to five tasks to each server. Develop a program in Java to implement a Least Connections load balancer for assigning the above tasks to servers. Consider each task to be represented by a unique integer. Display the workload on each server following the addition of each task.

Program 6 - Server Selection Using IP Hashing Technique

Scenario: Consider an input of 10 tasks which are to be assigned to three servers. Assign random and unique Ipv6 IP addresses to each task. Develop a program in Java to implement an IP hash load balancer for assigning the above tasks to servers. The hashing algorithm should map each possible IP address to one of the three servers. Consider each task to be represented by a unique integer. Display the workload on each server following the addition of each task.

Program 7 - Protocol Buffer-Based Student Record System

Understand the Protocol Buffers mechanism for serializing structured data. Understand the process of using a protocol compiler to compile a protocol buffer file. Create a protocol buffer file for a student record data structure and compile it using protocol for use with Java.

Program 8 - gRPC-Based Client–Server Application

Understand the gRPC Remote Procedure Call framework. Develop a program in Java to perform simple communication between a client and server using gRPC.

Program 9 - Horizontal Scaling Using MongoDB Sharding

Setup and deploy a MongoDB shard cluster and perform ranged sharding to understand horizontal scaling using sharding.

Program 10 - Deployment of a MongoDB Replica Set Cluster

Setup and deeploy a MongoDB replica set to understand horizontal scaling using replication.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Design scalable system architectures.
CO2	Implement cache replacement policies in scalable systems.
CO3	Implement load balancing schemes for large-scale systems.
CO4	Apply application layer protocols to implement scalable systems.
CO5	Apply horizontal database scaling techniques in distributed systems.

ASSESSMENT AND EVALUATION REGULATIONS OF AEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE Practical (A)	30	-	Marks awarded for Weekly Conduction of Experiments and Submission of Laboratory records, scaled to 30 marks.
CIE Practical Test (B)	20	-	Average of marks of Two tests, each conducted for 100 marks covering all experiments, scaled to 20
Total CIE Practical (A+B)	50	20	Sum of Marks of Experiments, Record and Test shall be CIE marks awarded for a maximum of 50.
SEE (C)	50	20	SEE is a lab exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Reference Books:

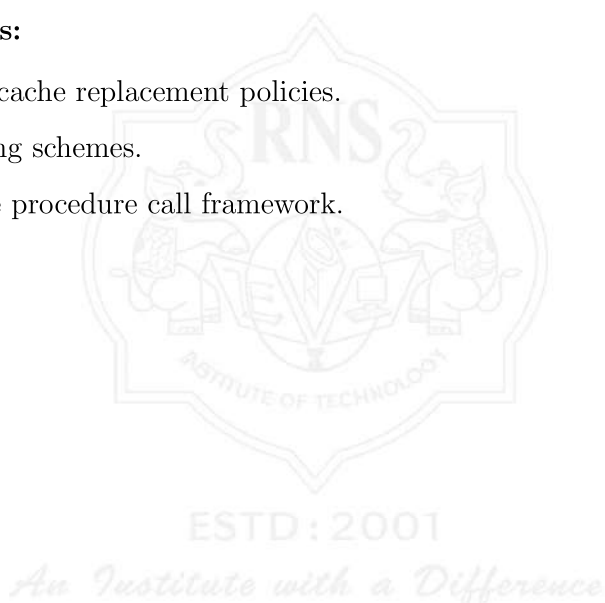
1. Martin Kleppmann, Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable and Maintainable Systems, 1st Edition, O'Reilly Media, Inc., 2017.

Activity Based Learning:

1. Case studies and group discussions about designing a URL shortener, a web crawler and a notification system.

Self Study Topics:

1. Caching and cache replacement policies.
2. Load balancing schemes.
3. gRPC remote procedure call framework.



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	COMPETITIVE PROGRAMMING		
Course Code	BCSL657E	CIE MARKS	50
(L : T : P)	(0:0:2)	SEE MARKS	50
Credits	1	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Implement and test solutions for competitive programming problems.
2. Use programming tools, libraries and online judges effectively.
3. Develop efficient algorithms and analyse performance through coding.
4. Practice real-time problem solving under contest constraints.

Prerequisites:

1. Basic knowledge of C/C++/Java/Python programming.
2. Understanding of control structures (loops, conditionals).
3. Familiarity with arrays, strings and functions.
4. Basic problem-solving and algorithmic thinking.

Program 1 - Activity Selection Using Greedy Approach

Scenario: You are given a set of activities, each with a start time and an end time. Develop a program to select the maximum number of activities that can be performed by a single person, assuming that a person can only work on one activity at a time. The selection strategy must follow a greedy approach: always choose the activity that finishes earliest among the remaining activities and does not conflict with previously selected ones.

Program 2 - Topological Sorting of Tasks in a DAG

Scenario: You are given a set of tasks represented as vertices of a Directed Acyclic Graph (DAG), along with their dependencies represented as directed edges. Develop a program to find a linear ordering of tasks using a topological sorting algorithm such that for every directed edge ($u \rightarrow v$), task u must be completed before task v .

Program 3 - Longest Common Substring and Subsequence Using Dynamic Programming

Scenario: a) Given two strings $s1 = \text{"ABCDGH"}$ and $s2 = \text{"ACDGHR"}$, write a program to determine the longest substring that is common to both strings using dynamic programming.
b) Given two strings $text1 = \text{"abcde"}$ and $text2 = \text{"ace"}$, write a program to find the length of their longest common subsequence using a 2D DP table.

Program 4 - Solving Linear Congruence Using Modular Inverse

Scenario: Given a linear congruence equation of the form: write a program to find the value of x using the concept of modular inverse. For this lab, solve the equation:

Program 5 - Naive Pattern Matching in Strings

Scenario: Given a text string and a pattern string, write a program to find all starting indices where the pattern occurs in the text. Use the Naive Pattern Matching approach, where the pattern is slid over the text one position at a time and characters are compared at each position. Text txt = "abcxabcdabcbcd" Pattern pat = "abcd"

Program 6 - Trie (Prefix Tree) Implementation for String Storage and Search

Scenario: Design and implement a Trie (Prefix Tree) to store a set of strings efficiently. Each node of the Trie should maintain links (children) to subsequent characters and a flag to indicate the end of a word. Insert the following words into the Trie: ["apple", "app", "application"] After building the Trie, perform the following operations: search("app"), search("apps") and startsWith("app")

Program 7 - Euclidean Distance Between Two Points in a 2D Plane

Scenario: Given two points P₁ (0,0) and P₂ (3,4), write a program to calculate the Euclidean distance between two points in a 2D plane using the distance formula.

Program 8 - Huffman Coding for Optimal Prefix Codes

Scenario: Given a set of characters along with their corresponding frequencies, write a program to construct a Huffman Tree and generate the optimal prefix codes for each character. Characters: {a, b, c, d, e, f} Frequencies: {5, 9, 12, 13, 16, 45}

Program 9 - Coin Change Problem: Minimum Coins and Counting Ways

Scenario: Given a set of coin denominations and a target amount, write a program to determine: a) The minimum number of coins required to make the given amount. b) The number of different ways to make the amount using the given denominations (considering unlimited supply of each coin).

Program 10 - Transitive Closure of a Directed Graph

Scenario: Given a directed graph represented using an adjacency matrix, write a program to determine its transitive closure.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Implement efficient programs to solve competitive programming problems.
CO2	Apply data structures and standard libraries in programming solutions.
CO3	Analyze problem-solving paradigms through coding practice.
CO4	Analyze graph-based and mathematical algorithms.

CO5	Analyze mathematical and string processing problems, including modular arithmetic, Euclidean distance and pattern matching.
-----	---

ASSESSMENT AND EVALUATION REGULATIONS OF AEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in components A, Band C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Reference Books:

1. Martin Kleppmann, Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable and Maintainable Systems, 1st Edition, O'Reilly Media, Inc., 2017.

Activity Based Learning:

1. Case studies and group discussions on designing a URL shortener, a web crawler and a notification system.

Self Study Topics:

1. Caching techniques
2. Load balancing
3. gRPC framework
4. MongoDB basics

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	MONITORING AND LOGGING SYSTEMS		
Course Code	BCSL657F	CIE MARKS	50
(L:T:P)	(0:0:2)	SEE MARKS	50
Credits	1	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The objectives of the course are to facilitate the learners to

1. Understand the fundamentals of system monitoring and logging mechanisms in distributed and cloud-based environments.
2. Develop skills to simulate and analyze system metrics such as CPU, memory, disk usage and service availability.
3. Implement logging frameworks and log analysis techniques to identify system behavior and issues.
4. Design and develop real-time monitoring applications using Python and Flask for system health tracking.
5. Apply techniques for anomaly detection, failure prediction and automated recovery (auto-healing) in monitored systems.

Prerequisites:

1. Basic programming knowledge on Python.
2. Basic idea of how a web application works (Flask basics).
3. Simple understanding of system resources like CPU and memory.

Program 1 - Simulated VM Performance Monitoring Dashboard

Scenario: A fast-growing startup, ScaleOps, hosts its application on multiple virtual machines (VMs). Recently, users have reported slow response times, but the engineering team lacks proper monitoring tools to analyze system performance. To address this issue, the team decides to build a lightweight monitoring solution that can simulate system metrics and provide visibility into VM health.

- a) Develop a program to simulate VM performance metrics such as CPU usage, memory usage and disk usage and display them in a formatted console dashboard view.
- b) Write a Python program to filter and display CPU utilization metrics from the simulated dataset generated in part (a).

Program 2 - Structured Logging System for Error Tracking

Scenario: You are working as a backend engineer at ShopSphere, an e-commerce platform. During peak sales, the system crashes intermittently, but the team cannot trace the issue due to the lack of proper logs. Senior engineers suggest implementing a structured logging system similar to Splunk to monitor and analyze system behavior.

- Develop a Python program to generate log messages with different severity levels such as INFO, WARNING, ERROR and CRITICAL. The program should display the log messages on the console and also store them in a log file.

Program 3 - System Memory and Service Availability Monitoring

Scenario: You are part of the infrastructure team at IntraServe, which manages internal tools used by employees. Recently, a memory leak caused a service outage and the issue went unnoticed until users reported it. To avoid such incidents, management asks the team to develop a basic monitoring tool.

- Develop a Python program to monitor system memory usage at regular intervals using libraries such as psutil and display the readings continuously.
- Develop a Python script to periodically check the availability of a given web service URL (or localhost) and display its status as UP or DOWN.

Program 4 - Lightweight Flask-Based System Metrics API

Scenario: At CloudBridge, DevOps engineers frequently log into servers to check system health. To simplify operations, the team plans to build a lightweight API (instead of tools like Grafana) that exposes system metrics and supports basic log analysis.

- Develop a Python-based web service using the Flask framework with an endpoint that displays the current system memory usage and measures and returns the response time for each incoming request.
- Implement a Python program to generate and store log messages, then analyze the log file to count total entries and ERROR-level logs.

Program 5 - Login Activity Tracking and Brute-Force Detection

Scenario: You are developing a login system for FinSecure Bank. Recently, there have been suspicious login attempts, possibly brute-force attacks. The security team needs visibility into login patterns.

- Implement a Python program to record user login activity, including username, timestamp and login status and detect cases where a user has more than three consecutive failed login attempts.
- Develop a Python program to generate login metrics such as total login attempts, total failed logins and login failure rate.

Program 6 - API Response Time Measurement and Audit Log Analysis

Scenario: You are working at APIHub, a platform providing public APIs. Clients complain about inconsistent response times and the security team suspects misuse of APIs.

- Develop a Python program to measure the response time of a given URL (public URL or localhost) over multiple attempts and display the minimum and maximum response times recorded.
- Implement a Python program to simulate audit logs as a list of dictionaries containing fields such as user, action, timestamp and status. Analyze the logs to detect suspicious activities such as unauthorized actions or repeated failures.

Program 7 - Automated Auto-Healing and Anomaly Detection System

Scenario: At CloudNova, downtime is costly. Instead of relying on engineers to manually fix issues, the company wants an automated recovery mechanism similar to modern cloud platforms.

- Develop a Python program to simulate an auto-healing mechanism that automatically performs predefined corrective actions, such as restarting a service or generating alerts, whenever critical or repeated error conditions are detected.
- Develop a Python program to analyze log data and detect abnormal patterns such as more than five failures within a fixed time interval and display an alert when such anomalies are detected.

Program 8 - IAM Log Simulation and Unauthorized Access Detection

Scenario: You are part of the cybersecurity team at SecureAccess Inc. The company suspects unauthorized login attempts from unfamiliar locations.

- Develop a Python program to simulate IAM logs (username, IP address, timestamp) and detect unauthorized access attempts such as login from unknown IP addresses.
- Write a program to classify login attempts based on time patterns and flag suspicious activity.

Program 9 - Cloud Cost Estimation and Abnormal Usage Spike Detection

Scenario: At CostWise Cloud, management is concerned about rising cloud expenses. Engineers need a system to understand how usage translates into cost and detect abnormal spikes.

- Develop a Python program to estimate cost based on system usage metrics such as CPU, memory and API usage using predefined pricing rules.
- Implement a program to dynamically adjust alert thresholds based on historical usage and detect abnormal spikes in resource consumption.

Program 10 - Microservice Dependency Failure Simulation and Log Pattern Analysis

Scenario: You are working on a microservices-based platform at ServiceMeshX. A failure in one service, such as authentication, can cause cascading failures in dependent services like database and payment systems.

- Implement a Python program to simulate dependent services using a dictionary (e.g., authentication to database to payment). If one service fails, display how the failure affects other dependent services.
- Develop a Python program to analyze log data and identify unusual patterns such as repeated errors from the same IP address using frequency counting techniques.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Apply monitoring techniques to measure system performance metrics such as CPU, memory and response time.
CO2	Implement logging mechanisms for system monitoring and analysis.
CO3	Monitor service availability to identify system failures.
CO4	Analyze logs to detect anomalies and potential security issues.
CO5	Apply monitoring concepts to generate alerts and perform automated recovery actions.

ASSESSMENT AND EVALUATION REGULATIONS OF AEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE Practical (A)	30	-	Marks awarded for Weekly Conduction of Experiments and Submission of Laboratory records, scaled to 20 marks.
CIE Practical Test (B)	20	-	Average of marks of Two tests, each conducted for 100 marks covering all experiments, scaled to 20
Total CIE Practical (A+B)	50	20	Sum of Marks of Experiments, Record and Test shall be CIE marks awarded for a maximum of 50.

SEE (C)	50	20	SEE is a lab exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE+SEE(A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Reference Books:

1. Anton Chuvakin, Kevin J. Schmidt and Christopher Phillips, Logging and Log Management: The Authoritative Guide to Understanding the Concepts Surrounding Logging and Log Management, 1st Edition, Syngress Elsevier, 2012.

Self Study topics:

1. Fundamentals of System Monitoring and Logging.
2. Log Analysis and Pattern Detection Techniques.
3. Cloud and DevOps Monitoring Tools.
4. Security Monitoring and Anomaly Detection.
5. Auto-Healing Systems and Failure Management.

Activity Based Learning:

1. Real-Time System Monitoring Activity.
2. Service Availability Testing Activity.
3. Security Scenario Simulation (Login Monitoring).

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	REINFORCEMENT LEARNING		
Course Code	BCSL657G	CIE MARKS	50
(L : T : P)	(0:0:2)	SEE MARKS	50
Credits	1	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand the fundamentals of Reinforcement Learning (RL), including agents, environments, states, actions, rewards and policies.
2. Apply basic RL algorithms such as greedy method, ϵ -greedy strategy, Value Iteration and Q-Learning to solve real-world engineering problems.
3. Analyze sequential decision-making problems in scenarios like robotics, traffic control, recommendation systems and inventory management using RL techniques.
4. Design simple RL-based models for optimization problems and evaluate their performance based on cumulative rewards and policy improvement.
5. Develop practical programming skills for implementing RL algorithms using basic tools and simulation environments.

Prerequisites:

1. Fundamentals of Python programming (variables, loops, functions, arrays, conditional statements).
2. Basic mathematics and probability concepts.
3. Understanding of matrices and graph representation.
4. Fundamentals of artificial intelligence and machine learning concepts.
5. Familiarity with simple simulation environments such as Grid World, Maze problems and game-based learning models.

Program 1 - Multi-Armed Bandit – Advertisement Selection

A company wants to choose the best advertisement among multiple options to maximize user clicks and revenue. Apply the ϵ -greedy algorithm to select the best advertisement and demonstrate how it maximizes cumulative reward over time.

Program 2 - Slot Machine Optimization

A gambler must choose among multiple slot machines, each having different reward probabilities. Implement and apply greedy and ϵ -greedy strategies to optimize rewards and compare their performance.

Program 3 - Grid World Navigation
A robot navigates a grid environment to reach a goal while avoiding obstacles. Apply value iteration to determine the optimal policy for the robot to reach the goal efficiently.
Program 4 - Maze Solving Robot
A robot must learn to escape from a maze by exploring different paths. Use Q-learning to enable the robot to learn the shortest path and demonstrate the learned policy.
Program 5 - Taxi Pickup Problem
A taxi operates in a city and must efficiently pick up and drop passengers. Apply Q-learning to optimize taxi operations for efficient passenger handling.
Program 6 - Inventory Management System
A shop owner needs to decide daily inventory levels to maximize profit while minimizing losses. Develop and apply a reinforcement learning model to determine optimal inventory levels under varying demand.
Program 7 - Traffic Signal Control
Traffic congestion at an intersection needs to be minimized using intelligent signal control. Apply Q-learning to dynamically control traffic signals and reduce vehicle waiting time.
Program 8 - Cart-Pole Balancing
A pole is attached to a cart and the system must keep the pole balanced. Implement a reinforcement learning approach to balance the pole by applying appropriate actions based on system state.
Program 9 - Tic-Tac-Toe Agent
An intelligent agent learns to play Tic-Tac-Toe against an opponent. Apply Q-learning to design an agent that learns to play optimally over time.
Program 10 - Recommendation System
A system recommends products to users based on their preferences and feedback. Develop a reinforcement learning-based recommendation system that adapts to user feedback.
Program 11 - Smart Home Energy Optimization
A smart home system controls appliances to reduce energy consumption. Apply reinforcement learning to optimize appliance usage while maintaining user comfort.
Program 12 - Elevator Scheduling System
An elevator system must efficiently serve multiple requests in a building. Implement a reinforcement learning algorithm to optimize scheduling and reduce waiting time.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the fundamental concepts of Reinforcement Learning, including agent, environment, state, action, reward and policy.
CO2	Implement basic Reinforcement Learning algorithms such as Greedy, ϵ -Greedy, Value Iteration and Q-Learning.
CO3	Analyze real-world problems using appropriate Reinforcement Learning techniques.
CO4	Develop Reinforcement Learning models for optimization problems.
CO5	Compare Reinforcement Learning approaches for engineering applications.

ASSESSMENT AND EVALUATION REGULATIONS OF AEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE Practical (A)	30	-	Marks awarded for Weekly Conduction of Experiments and Submission of Laboratory records, scaled to 30 marks.
CIE Practical Test (B)	20	-	Average of marks of Two tests, each conducted for 100 marks covering all experiments, scaled to 20.
Total CIE Practical (A+B)	50	20	Sum of Marks of Experiments, Record and Test shall be CIE marks awarded for a maximum of 50.
SEE (C)	50	20	SEE is a lab exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Reference Books:

1. Richard S. Sutton and Andrew G. Barto, Reinforcement Learning: An Introduction, 2nd Edition, MIT Press, 2018.
2. Maxim Lapan, Deep Reinforcement Learning Hands-On: Apply modern RL methods, with deep Q-networks, value iteration, policy gradients, TRPO and PPO, 1st Edition, Packt Publishing, 2018.
3. Csaba Szepesvári, Algorithms for Reinforcement Learning, 1st Edition, Morgan & Claypool Publishers, 2010.

Web links:

1. <https://youtu.be/2pWv7G0vuf0>
2. <https://www.youtube.com/watch?v=Qy-vum3GH-s>
3. https://www.youtube.com/watch?v=_VvnEu_2i2k
4. <https://www.youtube.com/watch?v=TCCjZe0y4Qc>
5. <https://www.youtube.com/watch?v=Nd1-UUMVfz4>
6. https://www.youtube.com/watch?v=PnHCvfgC_ZA
7. https://www.youtube.com/playlist?list=PLqYmG7hTraZBKeNJJJE_eyJHZ7XgBoAyb

Self Learning:

1. Case study: How AlphaGo uses Reinforcement Learning.
2. Exploration vs Exploitation dilemma in real-world decision making.
3. Case study: MDP in navigation problems.
4. Applications of reinforcement learning in self-driving cars and recommendation systems.

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	TIME SERIES ANALYSIS		
Course Code	BCSL657H	CIE MARKS	50
(L : T : P)	(0:0:2)	SEE MARKS	50
Credits	1	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The objectives of the course are to facilitate the learners to:

1. Understand and preprocess time series data by performing data cleaning, transformation, and visualization using appropriate tools.
2. Identify and analyze temporal patterns such as trend, seasonality and irregular variations in time series datasets.
3. Implement basic time series models including smoothing and autoregressive methods using programming libraries.
4. Develop and apply forecasting techniques to predict future values from time series data.
5. Evaluate and interpret forecasting results using suitable performance metrics and reproducible coding workflows.

Prerequisites:

1. Basic programming knowledge, preferably in Python and familiarity with simple data handling and visualization.
2. Fundamental understanding of statistics, including concepts such as mean, variance and correlation.

Program 1 - Data Acquisition & EDA

Scenario: A retail company collects daily sales data, but some dates are missing and there are sudden spikes during festivals.

Question: How would you preprocess this time series data and identify patterns such as trend, seasonality and anomalies using visualization and decomposition?

Program 2 - Stationarity Testing

Scenario: You are analysing monthly electricity consumption data that shows an increasing trend over time.

Question: How would you test whether the series is stationary using ADF and KPSS tests and what transformations would you apply if it is non-stationary?

Program 3 - AR, MA, ARMA Models

Scenario: A financial analyst wants to model stock price fluctuations that appear random but show some autocorrelation.

Question: How would you choose between AR, MA and ARMA models and how would you evaluate which model fits best?

Program 4 - ARIMA & SARIMA

Scenario: Airline passenger data shows both trend and strong seasonal patterns.

Question: How would you build ARIMA and SARIMA models for forecasting and how would you compare their forecasting accuracy?

Program 5 - Exponential Smoothing (ETS)

Scenario: A company wants to forecast demand for a product with clear trend and seasonal effects.

Question: How would you apply Simple, Holt and Holt-Winters models and how would you compare their performance with ARIMA?

Program 6 - ARIMAX (with External Variables)

Scenario: You are predicting ice cream sales using temperature as an external factor.

Question: How would you incorporate temperature into a time series model and how would you interpret its impact on sales?

Program 7 - Machine Learning Forecasting

Scenario: An e-commerce company wants to predict daily website traffic using past data.

Question: How would you create lag features and rolling statistics and which machine learning model would you use for forecasting?

Program 8 - Deep Learning (RNN/LSTM)

Scenario: A company wants to forecast stock prices using complex patterns in historical data.

Question: How would you design an LSTM model using sliding windows and how would you evaluate its performance compared to traditional models?

Program 9 - Frequency Domain Analysis

Scenario: A sensor records vibration data from a machine and you suspect periodic faults.

Question: How would you use Fourier Transform to identify dominant frequencies and detect periodic behavior?

Program 10 - Anomaly & Change Point Detection

Scenario: A bank monitors transaction data and wants to detect fraudulent activities and sudden behaviour changes.

Question: How would you identify anomalies and change points and how would you evaluate the effectiveness of your detection method?

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Apply techniques for time series data acquisition, preprocessing, exploratory analysis and stationarity testing.
CO2	Implement classical time series models such as AR, MA, ARMA, ARIMA and SARIMA for forecasting.
CO3	Develop forecasting models using exponential smoothing, state space models and regression with exogenous variables.
CO4	Analyze advanced techniques, including spectral analysis, machine learning and deep learning models for time series prediction.
CO5	Design systems for anomaly detection, change point detection and real-time monitoring in time series data.

ASSESSMENT AND EVALUATION REGULATIONS OF AEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE Practical (A)	30	-	Marks awarded for Weekly Conduction of Experiments and Submission of Laboratory records, scaled to 20 marks.
CIE Practical Test (B)	20	-	Average of marks of Two tests, each conducted for 100 marks covering all experiments, scaled to 20
Total CIE Practical (A+B)	50	20	Sum of Marks of Experiments, Record and Test shall be CIE marks awarded for a maximum of 50.
SEE (C)	50	20	SEE is a lab exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A + B + C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.

The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.

Textbooks:

1. Peter J. Brockwell and Richard A. Davis, Introduction to Time Series and Forecasting, 3rd Edition, Springer, 2016.
2. George E. P. Box, Gwilym M. Jenkins, Gregory C. Reinsel and Greta M. Ljung, Time Series Analysis: Forecasting and Control, 5th Edition, John Wiley & Sons, Inc., 2015.

Reference Books:

1. Rob J. Hyndman and George Athanasopoulos, Forecasting: Principles and Practice, 3rd Edition, OTexts, 2021.

Self Study topics:

1. Time Series Data Sources and Applications: A smart city analyst works with time-based data from traffic, weather and energy systems. Explore real-world datasets and understand their structure and applications in forecasting.
2. Advanced Time Series Visualization and Decomposition Techniques: A company analyses sales trends to identify seasonal patterns and long-term behaviour. Study visualization and decomposition methods to interpret temporal data effectively.
3. Model Selection and Hyper parameter Tuning in Forecasting: A forecasting team must choose the best model for demand prediction. Investigate different time series models and techniques for tuning parameters to improve accuracy.
4. Handling Missing Data, Noise and Outliers in Time Series: IoT sensor data often contains missing values and irregularities. Study pre-processing techniques to clean data and enhance model reliability.
5. Real-Time Time Series Analysis and Deployment: A financial system monitors stock prices continuously for anomalies. Explore how time series models are deployed for real-time analysis and decision-making.

Activity Based Learning:

1. Dataset Exploration Activity – Students collect and analyze real-world time series datasets to identify patterns such as trend, seasonality and irregular variations.
2. Forecasting Model Challenge – Students build and compare different forecasting models (statistical or machine learning) on a given dataset and evaluate their performance.
3. Anomaly Detection Case Study – Students analyze a time series dataset to detect anomalies or unusual patterns and design a basic alert mechanism.

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	SERVERLESS COMPUTING		
Course Code	BCSL657I	CIE MARKS	50
(L : T : P)	(0:0:2)	SEE MARKS	50
Credits	1	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Introduce the concepts and architecture of serverless computing and its role in modern cloud computing environments.
2. Understand Function as a Service (FaaS) and event-driven application development models.
3. Provide practical knowledge of serverless platforms such as AWS Lambda, Azure Functions and Google Cloud Functions.
4. Develop skills to design, deploy and manage scalable serverless applications.
5. Analyze security, orchestration and real-world architectures used in serverless systems.

Program 1 - Installing the Development Suite

Set up Visual Studio Code, Node.js and Postman on your local workstation to prepare for serverless development.

Program 2 - Configuring Cloud SDKs

Install and authenticate the AWS Node.js SDK, Azure Node.js SDK and Google Cloud SDK (gcloud CLI) to interact with cloud services via the command line.

Program 3 - The "Hello World" Lambda

Create your first function using the AWS Console, configure basic settings (memory and timeout) and test it using a sample event template.

Program 4 - Lambda Versioning and Aliases

Practice publishing new versions of a function and creating an alias (such as PROD or DEV) to manage traffic shifting between different code iterations.

Program 5 - S3 Event Trigger

Configure an S3 trigger so that uploading a file to a specific bucket automatically invokes a Lambda function to process that file.

Program 6 - Building with AWS SAM

Develop a serverless application locally using the AWS Serverless Application Model (SAM), test it in your local environment and deploy it to the cloud.

Program 7 - Serverless Orchestration with Step Functions

Create a state machine to orchestrate multiple Lambda functions in sequential, branching, or parallel steps to handle complex workflows.

Program 8 - HTTP-Triggered Azure Function

Use the Azure Portal to create a function that responds to HTTP requests and test it using the built-in console and Postman.

Program 9 - Azure Blob Storage Bindings

Implement a function that uses Input/Output bindings to automatically read from or write data to Azure Blob Storage without writing boilerplate integration code.

Program 10 - Timer-Triggered Tasks

Create a function with a Timer trigger (using CRON expressions) to execute tasks at specific intervals, such as daily database cleanups.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the fundamental concepts of serverless computing, including event-driven architecture, Function as a Service (FaaS), benefits and limitations.
CO2	Configure development environments, tools and SDKs such as Visual Studio Code, Node.js and Postman for serverless applications.
CO3	Develop serverless functions using AWS Lambda, including configuration, triggers, events and security mechanisms.
CO4	Implement serverless solutions using Azure Functions, including triggers, bindings, hosting plans and platform features.
CO5	Design serverless applications and workflows on AWS using deployment and orchestration services such as Step Functions.

ASSESSMENT AND EVALUATION REGULATIONS OF AEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE Practical (A)	30	-	Marks awarded for Weekly Conduction of Experiments and Submission of Laboratory records, scaled to 20 marks.
CIE Practical Test (B)	20	-	Average of marks of Two tests, each conducted for 100 marks covering all experiments, scaled to 20

Total CIE Practical (A+B)	50	20	Sum of Marks of Experiments, Record and Test shall be CIE marks awarded for a maximum of 50.
SEE (C)	50	20	SEE is a lab exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A + B + C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Kuldeep Chowhan, Hands-On Serverless Computing: Build, run and orchestrate serverless applications using AWS Lambda, 1st Edition, Packt Publishing, 2018.

Reference Books:

1. Peter Sbarski, Serverless Architectures on AWS, 1st Edition, Manning Publications, 2017.
2. Sheen Brisals and Luke Hedger, Serverless Development on AWS: Building Enterprise-Scale Serverless Applications, 1st Edition, O'Reilly Media, 2024.
3. Brian Zambrano, Serverless Design Patterns and Best Practices: Build, secure and deploy enterprise ready serverless applications with AWS, 1st Edition, Packt Publishing, 2018.
4. James Beswick, Building Serverless Applications with AWS SAM, 1st Edition, O'Reilly Media, 2021.

Web Links:

1. <https://aws.amazon.com/serverless/>
2. <https://docs.aws.amazon.com/lambda/>
3. <https://cloud.google.com/serverless>
4. <https://learn.microsoft.com/azure/azure-functions/>
5. <https://www.serverless.com/framework/docs/>

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	SCALA AND JULIA		
Course Code	BCSL657J	CIE MARKS	50
(L : T : P)	(0:0:2)	SEE MARKS	50
Credits	1	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand the fundamental concepts, syntax and programming paradigms of Scala and Julia.
2. Apply problem-solving skills using control structures, functions and recursion.
3. Analyse and manipulate data using arrays, strings, collections and file handling techniques.
4. Design modular programs using object-oriented and functional programming concepts.

Prerequisites:

1. Basic Programming Knowledge: Familiarity with any programming language (C/Python/Java), including variables, data types and input/output operations.
2. Control Structures & Functions: Understanding of conditional statements, loops and basic function usage for problem solving.
3. Fundamentals of Data Handling: Basic knowledge of arrays and strings.
4. Computer & Mathematical Basics: Ability to compile and run programs, along with basic arithmetic and logical reasoning skills.

Program 1 - Student Grade Calculator (Using Case Class)

Scenario: A college plans to automate student evaluation. For each student, the system must store the name and marks in three subjects, compute the total and average and assign a grade based on performance criteria (A, B, C, Fail). Design a Scala program using appropriate constructs such as case classes to implement this system.

Program 2 - Data Processing using Collections

A retail company maintains a list of daily sales figures and wants to analyze the data efficiently. The system should generate a new list containing squares of all values, filter out even numbers and compute the total sales using suitable functional programming techniques in Scala.

Program 3 - Smart Calculator

Scenario: A calculator application is required to perform different arithmetic operations based on the operator entered by the user. The system should identify the operation dynamically and execute it accordingly using pattern matching in Scala.

Program 4 - File Analysis Tool

Scenario: An organization needs a tool to analyze text files for basic statistics. The system should read a file and determine the total number of lines, words and characters present using file handling techniques in Scala.

Program 5 - Banking System

Scenario: A bank wants to manage customer accounts digitally. Each account should store account details and allow operations such as deposit, withdrawal and balance enquiry. Design a Scala program using classes and objects to simulate this banking system.

Program 6 - Matrix Operations in Engineering

Scenario: In engineering computations, matrices are frequently used for solving problems. A system is required to accept two matrices and perform both addition and multiplication operations, displaying the results using Julia programming.

Program 7 - Multi-Operation Function

Scenario: A scientific application requires a function that can process two input values and simultaneously return multiple results such as sum, difference and product. Implement this functionality using Julia's ability to return multiple values.

Program 8 - Data Analysis System

Scenario: A data analyst needs to process a dataset to extract meaningful insights. The system should compute statistical measures such as mean, median, maximum and minimum values for a given set of numbers using Julia.

Program 9 - Recursive Computation

Scenario: A computational system needs to generate a sequence where each term depends on previous terms. Implement a solution using recursion in Julia to generate the Fibonacci series for a given number of terms.

Program 10 - Data Visualization

Scenario: A research team wants to visualize mathematical relationships graphically. The system should generate a plot for a mathematical function (such as $y = x^2$) to help in analyzing trends and patterns using Julia.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Demonstrate understanding of syntax, data types and basic programming constructs in Scala and Julia.
-----	--

CO2	Apply control structures, functions and recursion to solve computational problems.
CO3	Analyze and process data using arrays, strings, collections and file handling techniques.
CO4	Design and develop modular programs using object-oriented and functional programming concepts.
CO5	Implement problem-solving solutions using Scala and Julia for real-world applications.

ASSESSMENT AND EVALUATION REGULATIONS OF AEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE Practical (A)	30	-	Marks awarded for Weekly Conduction of Experiments and Submission of Laboratory records, scaled to 20 marks.
CIE Practical Test (B)	20	-	Average of marks of Two tests, each conducted for 100 marks covering all experiments, scaled to 20
Total CIE Practical (A+B)	50	20	Sum of Marks of Experiments, Record and Test shall be CIE marks awarded for a maximum of 50.
SEE (C)	50	20	SEE is a lab exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A + B + C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Reference Books:

1. Odersky, M., Spoon, L., & Venners, B. 2016. Programming in Scala 3rd Edition. Artima Press.
2. Nagar, S. 2017. Beginning Julia Programming: For Engineers and Scientists. Apress.

Web Links:

1. <https://www.coursera.org/programs/faculty-learning-programtaue8/learn/scalafunctional-programming?source=search>
2. <https://www.coursera.org/videos/juliaprogramming/aUI5C?query=julia+basics&source=search>

Self Study Topics:

You can explore advanced concepts such as higher-order functions, error handling, collections in Scala and data analysis, visualization and packages in Julia.

Activity Based Learning:

1. Develop grade analyzers to process and evaluate student performance.
2. Build banking simulation systems to perform financial transactions.
3. Create data analysis systems to organize and analyze datasets.
4. Design data visualization tasks to present information using charts and graphs.
5. Complete a group mini-project to apply programming concepts to real-world scenarios.

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	INDIAN KNOWLEDGE SYSTEMS		
Course Code	BIKS608	CIE MARKS	100
(L : T : P)	(1:0:0)	SEE MARKS	—
Credits	—	EXAM HOURS	2

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. To facilitate the students with the concepts of Indian traditional knowledge and to make them understand the Importance of roots of knowledge system.
2. To make the students understand the traditional knowledge and analyze it and apply it to their day-to-day life.

Module-1

Introduction to Indian Knowledge Systems (IKS): Overview, Importance of ancient knowledge, The IKS corpus, Vedic Corpus, Philosophical systems, Character scope and importance, traditional knowledge vis-a-vis indigenous knowledge, traditional knowledge vs. western knowledge.

Module-2

Fundamental concepts for science and technology: Linguistics, Numbers and measurements- Mathematics, Chemistry, Physics, Art, Astronomy, Astrology, Crafts and Trade in India and Engineering and Technology.

Module-3

Traditional Knowledge in Professional Domain: Town planning and architecture, Construction, Health, wellness and Psychology-Medicine, Agriculture, Governance and Public administration.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Provide an overview of the concept of the Indian Knowledge System and its importance.
CO2	Appreciate the need for and importance of protecting traditional knowledge.
CO3	Recognize the relevance of Traditional knowledge in different domains.
CO4	Establish the significance of Indian Knowledge systems in the contemporary world.

ASSESSMENT AND EVALUATION REGULATIONS OF MC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE - IA Tests (A)	60	-	Sum of Two Internal Assessment Tests (MCQs) each conducted for 30 marks.
Continuous and Comprehensive Evaluation (B)	40	-	Any two Assessment methods as per 24RNBE4.2, scaled to 40 marks.
Total CIE (A+B)	100	40	CIE Evaluation is done for a maximum of 100 marks.
The minimum marks to be secured in CIE is 40 (40%) out of allotted 100 marks to Pass.			

Textbooks:

1. B. Mahadevan, Vinayak Rajat Bhatand Nagendra Pavana R. N., Introduction to Indian Knowledge System: Concepts and Applications, 1st Edition, PHI Learning Private Ltd., 2022.
2. M. Morris Mano, Digital Logic and Computer Design, 1st Edition, PHI Learning, 2008.

Reference Books:

1. Amit Jha, Traditional Knowledge System in India, 1st Edition, Atlantic Publishers and Distributors (P) Ltd., 2009.
2. Kapil Kapoor and Avadesh Kumar Singh, Knowledge Traditions and Practices of India, Vol. 1, 1st Edition, D.K. Printworld (P) Ltd., 2005.

Web Links:

1. <https://www.youtube.com/watch?v=LZP1StpYEPm>
2. <http://nptel.ac.in/courses/121106003/>
3. <https://11nq.com/frgzznh>
4. https://www.wipo.int/pressroom/en/briefs/tk_ip.html
5. https://unctad.org/system/files/official-document/ditcted10_en.pdf
6. http://nbaindia.org/uploaded/docs/traditionalknowledge_190707.pdf
7. https://unfoundation.org/what-we-do/issues/sustainable-development-goals/?gclid=EAIaIQobChMImp-Jtb_p8gIVTeN3Ch27LAMPEAAYASAAEgIm1vD_BwE

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	NATIONAL SERVICE SCHEME		
Course Code	BNSK659	CIE MARKS	100
(L : T : P)	(0:0:2)	SEE MARKS	—
Credits	—	EXAM HOURS	—

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Identify the needs and problems of the community and involve in problem solving.
2. Develop a sense of social and civic responsibility and utilize knowledge in finding practical solutions to individual and community problems.
3. Develop competence required for group living and sharing of responsibilities and gain skills in mobilising community participation to acquire leadership qualities and democratic attitudes.

Module-1

Contribution to national-level initiatives of the Government of India, such as Digital India, Skill India, Swachh Bharat, Atmanirbhar Bharat, Make in India, the Mudra Scheme and other skill development programs.

Module-2

Spreading public awareness through rural outreach programs. (minimum 5 programs).

Module-3

Innovative approaches to assist in the marketing of rural produce

Module-4

Creating awareness on rainwater harvesting in Urban areas.

Module-5

Creating awareness on digital payment frauds to locals.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Identify and analyze community needs and actively participate in solving social issues through NSS activities.
CO2	Demonstrate social and civic responsibility by engaging in national initiatives and rural outreach programs.
CO3	Develop leadership qualities, teamwork skills and the ability to mobilize community participation effectively.

CO4	Apply innovative approaches to address real-world problems such as rural development, environmental sustainability and digital awareness.
-----	---

ASSESSMENT AND EVALUATION REGULATIONS OF MC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE - IA Tests (A)	60	-	Sum of Two Internal Assessment Tests (MCQs) each conducted for 30 marks.
Continuous and Comprehensive Evaluation (B)	40	-	Any two Assessment methods as per 24RNBE4.2, scaled to 40 marks.
Total CIE (A+B)	100	40	CIE Evaluation is done for a maximum of 100 marks.
The minimum marks to be secured in CIE is 40 (40%) out of allotted 100 marks to Pass.			

Reference Books:

1. NSS Course Manual, Published by NSS Cell, VTU Belagavi.
2. Government of Karnataka, NSS Cell, Activity Reports and Manual.
3. Government of India, NSS Cell, Activity Reports and Manual.

Web Links:

1. National Service Scheme (NSS), Ministry of Youth Affairs and Sports.
<https://nss.gov.in>
2. MyGov Pledge Portal, Government of India.
<https://pledge.mygov.in>

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	PHYSICAL EDUCATION		
Course Code	BPEK659	CIE MARKS	100
(L : T : P)	(0:0:2)	SEE MARKS	—
Credits	—	EXAM HOURS	—

COURSE LEARNING OBJECTIVES:

The purpose of the course is to facilitate the learner to:

1. Understand the fundamental concepts of Physical Education, games and sports skills, Health, Nutrition and Fitness.
2. Familiarization of health-related Exercises, Sports for overall growth and development.
3. Participate in the competition at regional/state / national / international levels.
4. Create consciousness among the students on Health, Fitness and Wellness in developing and maintaining a healthy lifestyle.

Module-1: Title: Athletics - Track Events - Relays **3 Hours**

Relays

1. Relay Race: Starting, Baton Holding/Carrying, Baton Exchange in between zone and Finishing.

Module-2: Title: Athletics - Triple Jumps **3 Hours**

Triple Jump

- A) Approach Run
- B) Take-off
- C) Step
- D) Flight in the Hop
- E) Jump and Landing

Module-3: Title: Athletics - Throws **3 Hours**

Javelin Throw

- A) Grip
- B) Carry and Recovery (3/5 Impulse stride)
- C) Release

Module-4: Title: CRICKET

3 Hours

A. Fundamental skills

1. Batting - Forward Defense Stroke, Backward Defense Stroke, Off Drive, On Drive, Straight Drive, Cover Drive, Square Cut.
2. Bowling - Out-swing, In-swing, Off Break, Leg Break and Googly.
3. Fielding: Catching - The High Catch, The Skim Catch, The Close Catch and throwing at the stumps from different angles. Long Barrier and Throw, Short Throw, Long Throw, Throwing on the Turn.
4. Wicket Keeping

B. Rules and their interpretation and duties of the officials

Module-5: Title: BASEBALL

3 Hours

A. Fundamental skills

1. Player Stances – walking, extending walking, L stance, cat stance.
2. Grip – standard grip, choke grip.
3. Batting – swing and bunt. Pitching.
4. Baseball: slider, fast pitch, curve ball, drop ball, rise ball, change up, knuckle ball, screw ball.

B. Rules and their interpretation and duties of the officials

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Demonstrate the techniques of relay races, including starting, baton holding, exchange within the zone and finishing, along with basic skills of javelin throw.
CO2	Apply the fundamental techniques of triple jump, including approach run, take-off, hop, step, jump, flight and landing.
CO3	Develop and perform javelin throw techniques, including grip, carry, impulse stride, release and recovery with proper coordination.
CO4	Exhibit fundamental skills of cricket, including batting, bowling, fielding, wicket keeping and understand rules and duties of officials.
CO5	Demonstrate basic skills and techniques in baseball, including player stances, grip, batting, pitching and understand rules and officiating.

ASSESSMENT AND EVALUATION REGULATIONS OF MC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	60	—	Sum of Two Internal Assessment Tests (MCQs) each conducted for 30 marks.
Project (B)	40	—	—
Total CIE (A+B)	100	40	CIE Evaluation is done for a maximum of 100 marks.
The minimum marks to be secured in CIE is 40 (40%) out of 100 marks allotted to Pass.			

Textbooks:

1. Health and Physical Education by V.K. Sharma
2. Introduction to Physical Education, Fitness and Sport by Daryl Siedentop
3. Physical Education Activities, Sports and Games: The Origin and Background, Rules of Play, Teaching Hints and Fundamentals by Louis E. Means

Reference Books:

1. Saraswati Health and Physical Education Textbook
2. Foundations of Physical Education, Exercise Science and Sport by Deborah A. Wuest

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VI

Course Title	YOGA		
Course Code	BYOK659	CIE MARKS	100
(L : T : P)	(0:0:2)	SEE MARKS	—
Credits	—	EXAM HOURS	—

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Physical, Mental and Spiritual objectives of Yoga
2. Explain the meaning of Yoga
3. Discuss the importance of Yoga and its scope
4. Describe the aims and objectives of Yoga
5. Clarify the misconceptions on Yoga

Module-1: Suryanamaskara

Suryanamaskara - Revision of practice 12 count 8 rounds

- a) Suryanamaskar prayer.
- b) Meaning, need & Importance.
- c) Benefits of Suryanamaskara.

Module-2: Asanas

Need & importance of different Asanas, their meaning by name, technique, precautionary measures and benefits of each asana.

Sitting:

1. Bakasana
2. Hanumanasana
3. Yogamudra in Vajrasana

Standing:

1. Vatayanasana
2. Garudasana

Module-3: Asanas

Need and importance of Asanas. Different types, meaning by name, technique, precautionary measures and benefits of each asana.

Prone Line:

1. Bhujangasana
2. Shalabhasana

Supine Line:

1. Utthita Dvipadasana
2. Ardha Halasana

Balancing:

1. Sheershasana

Module-4: Pranayama

Kapalabhati – Revision of Kapalabhati practice (80 strokes/minute, 3 rounds).

Module-5: Pranayama

Pranayama:

1. Bhastrika
2. Bhramari

Meaning, need, importance, different types, technique, precautionary measures and benefits of each Pranayama.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the meaning, objectives, scope and importance of Yoga, including its physical, mental and spiritual benefits and misconceptions.
CO2	Demonstrate Suryanamaskara (12 counts, 8 rounds) with correct technique, prayer and understanding of its need and benefits.
CO3	Perform sitting and standing Asanas (Bakasana, Hanumanasana, Yogamudra in Vajrasana, Vatayanasana, Garudasana) with proper technique, precautions and benefits.
CO4	Execute prone, supine and balancing Asanas (Bhujangasana, Shalabhasana, Utthita Dvipadasana, Ardha Halasana, Sheershasana) with correct body alignment and safety measures.

CO5	Practice Pranayama techniques (Kapalabhati, Bhastrika, Bhramari) with proper method, breathing control and understanding of their importance and health benefits.
------------	---

ASSESSMENT AND EVALUATION REGULATIONS OF MC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	60	—	Sum of Two Internal Assessment Tests (MCQs) each conducted for 30 marks.
Project (B)	40	—	—
Total CIE (A+B)	100	40	CIE Evaluation is done for a maximum of 100 marks.
The minimum marks to be secured in CIE is 40 (40%) out of 100 marks allotted to Pass.			

Textbooks:

1. The Yoga Sutras of Patanjali by Sri Swami Satchidananda
2. Light on Yoga by B.K.S. Iyengar

Reference Books:

1. The Heart of Yoga by T.K.V. Desikachar
2. Asana Pranayama Mudra Bandha by Swami Satyananda Saraswati

VII SEMESTER SUBJECTS

ESTD: 2001
An Institute with a Difference

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VII

Course Title	HIGH PERFORMANCE COMPUTING		
Course Code	BCS701	CIE MARKS	50
(L : T : P) + SL	(3:0:2)+(45 Hours/Sem)	SEE MARKS	50
Credits	4	EXAM HOURS	3

COURSE LEARNING OBJECTIVES:

The purpose of the course is to facilitate the learner to:

1. Understand High Performance Computing (HPC) system architectures and computational models.
2. Learn parallel programming paradigms including Shared Memory and Message Passing.
3. Learn GPU programming concepts using Compute Unified Device Architecture (CUDA) and OpenCL.
4. Design and implement compute-intensive applications on HPC platforms.

Module-1	9 Hours
Modern Processor: Stored-program architecture, General-purpose cache-based micro-processor architecture: pipelining, superscalarity, SIMD, Memory hierarchies, multicore processors, Multithreaded processors. Vector processors: Design principles, maximum performance estimates, programming for vector architectures. Textbook 1: Chapter 1 (1.1, 1.2.3–1.2.5, 1.3–1.4, 1.6.1–1.6.3)	
Module-2	9 Hours
Parallel Computers: Taxonomy of parallel computing paradigms, Shared-memory systems (UMA, ccNUMA), Distributed-memory systems, Hybrid systems, Interconnection networks. Cluster Computing: A Cluster Computer and its Architecture, Commodity Components for Clusters, Cluster Middleware, Resource Management and Scheduling. Textbook 1: Chapter 4 (4.1–4.5) Textbook 2: Chapter 1 (1.5, 1.7, 1.9–1.10)	
Module-3	9 Hours
Basics of Parallelization: Why Parallelize? Parallelism, Parallel Scalability: Simple Scalability Laws, Parallel Efficiency, Load Imbalance. Constructing Scalable Services: Environment, Resource Sharing, Resource Sharing Enhanced Locality. Textbook 1: Chapter 5 (5.1 - 5.2, 5.3.3 - 5.3.4, 5.3.9) Textbook 2: Chapter 3 (3.2 - 3.4)	

Module-4	9 Hours
Shared-memory Parallel Programming with OpenMP: Short introduction to OpenMP, Parallel execution, Data scoping, OpenMP work-sharing for loops, Synchronization, Reductions, Loop scheduling, Efficient OpenMP programming, Profiling OpenMP programs, Performance pitfalls. Programming Environments and Tools: Threads, Message Passing Systems (MPI and PVM), Distributed Shared Memory (DSM) Systems, Parallel Debuggers and Profilers, Performance Analysis Tools, Cluster Administration Tool. Textbook 1: Chapter 6 (6.1, 6.1.1 - 6.1.6), Chapter 7 (7.1 - 7.2) Textbook 2: Chapter 1 (1.11.1 - 1.11.6)	
Module-5	9 Hours
Distributed-memory Parallel Programming with MPI: Message passing, Introduction to MPI, A simple example, Messages and point-to-point communication, Collective communication, Non-blocking point-to-point communication, Virtual topologies. Hybrid Parallelization with MPI and OpenMP: Basic MPI/OpenMP programming models, Vector mode implementation, Task mode implementation, Case study: Hybrid Jacobi solver, MPI taxonomy of thread interoperability, Hybrid decomposition and mapping. Textbook 1: Chapter 9 (9.1 - 9.2), Chapter 11 (11.1 - 11.3)	

PRACTICAL COMPONENT OF IPCC

PART A
1. Develop MPI program for message passing In a distributed computing environment, multiple processors need to exchange intermediate results to complete a scientific computation. Each processor computes partial data and sends it to another processor for further processing. Write an MPI program to: 1. Initialize MPI environment. 2. Send data from Process 0 to Process 1. 3. Receive the data at Process 1 and display it. 4. Finalize the MPI environment. In a weather forecasting system, sensor nodes collect temperature data and send it to a central processor.

Write an MPI program to:

1. Distribute temperature values among processes.
2. Send data from worker processes to master.
3. Receive and display all values at master.
4. Terminate MPI execution.

2. Implement MPI-based parallel reduction

Write an MPI program to:

1. Distribute an array among processes.
2. Compute local sum.
3. Use MPI_Reduce to compute global sum.
4. Display result.

Write an MPI program to:

1. Assign transaction values.
2. Compute local maximum.
3. Use MPI_Reduce.
4. Display result.

3. Compare sequential and parallel performance

Write a program to:

1. Implement sequential matrix multiplication.
2. Implement parallel version.
3. Measure execution time.
4. Compare results.

Write a program to:

1. Perform sequential sorting.
2. Perform parallel sorting.
3. Measure execution time.
4. Compare performance.

4. Analyze speedup and efficiency using Amdahl's Law

Write a program to:

1. Accept serial fraction and processors.
2. Compute speedup.
3. Calculate efficiency.
4. Display results.

Write a program to:

1. Take processor counts.
2. Compute speedup.
3. Calculate efficiency.
4. Display results.

PART B

5. HPC architectures application

Write a program to:

1. Demonstrate shared memory.
2. Simulate distributed computation.
3. Demonstrate GPU acceleration.
4. Compare performance.

Scenario Based Question – 2

Write a program to:

1. Implement shared memory processing.
2. Distribute workload.
3. Demonstrate GPU.
4. Compare results.

6. Parallel programs on shared memory systems

Write a program using OpenMP to:

1. Initialize array.
2. Perform parallel computation.
3. Use OpenMP directives.
4. Display result.

Write a program to:

1. Divide dataset.
2. Compute partial averages.
3. Combine results.
4. Display output.

7. Parallel programs for distributed memory (Matrix Multiplication)

Write an MPI program to:

1. Distribute matrices.
2. Perform multiplication.
3. Gather results.
4. Display matrix.

Write an MPI program to:

1. Divide matrix rows.
2. Perform multiplication.
3. Collect results.
4. Display output.

8. CUDA-based GPU acceleration

Write a CUDA program to:

1. Allocate GPU memory.
2. Transfer data.
3. Execute kernel.
4. Display result.

Write a CUDA program to:

1. Load image to GPU.
2. Perform transformation.
3. Transfer back.
4. Display output.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Analyze High Performance Computing (HPC) architectures, computational models and parallelism paradigms for compute-intensive applications.
CO2	Implement shared-memory (OpenMP) and distributed-memory (MPI) parallel solutions for compute-intensive problems.
CO3	Analyze message passing, synchronization strategies and inter-process communication mechanisms in parallel systems.
CO4	Implement GPU-based and hybrid parallel applications using CUDA/OpenCL and OpenMP.
CO5	Analyze the performance of GPU-based and hybrid parallel applications using benchmarking techniques.

ASSESSMENT AND EVALUATION REGULATIONS OF IPCC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	20	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 20 marks.
Continuous and Comprehensive Evaluation (B)	10	-	Any one assessment method as per 24RNBE4.2 scaled to 10 marks.
Total CIE Theory (A+B)	30	12	Marks of Tests and Assignments awarded for a maximum of 30.
CIE Practical	12	-	Marks awarded for weekly conduction of experiments and submission of laboratory records, scaled to 12 marks.
CIE Practical Test	8	-	Average of marks of two tests, each conducted for 100 marks covering all experiments, scaled to 8.
Total CIE Practical (C)	20	8	Marks of experiments, record and test awarded for a maximum of 20 marks.
CIE: Theory + Practical (A+B+C)	50	20	CIE evaluation is done for a maximum of 50 marks.
SEE (D)	50	18	SEE is a theory exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall be 3 hours.
CIE+SEE (A+B+C+D)	100	40	Final marks shall be the sum of marks scored in components A, B, C and D.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			
The laboratory component of the IPCC shall be for CIE only. However, in SEE, the questions from the laboratory component shall be included in their respective modules only.			

Textbooks:

1. Georg Hager and Gerhard Wellein, Introduction to High Performance Computing for Scientists and Engineers, 1st Edition, CRC Press Taylor & Francis Group, 2011.

2. Rajkumar Buyya Edition, High Performance Cluster Computing: Architectures and Systems, Volume 1, 1st Edition, Prentice Hall PTR, 1999.

Reference Books:

1. Kai Hwang & Naresh Jotwani, Advanced Computer Architecture: Parallelism, Scalability, Programmability, 3rd Edition, McGraw Hill Education, 2015.

Web Links:

1. <https://nptel.ac.in/courses/106108055>
2. <https://nptel.ac.in/courses/106105033>

Self Study topics:

1. Roofline Performance Model: Analysis of arithmetic intensity and memory bandwidth limits to identify performance bottlenecks in parallel applications.
2. Advanced Cache Coherence Mechanisms: Study of MESI/MOESI and directory-based protocols used in modern multicore and manycore processors.
3. GPU Memory Optimization Techniques: Efficient utilization of shared memory, registers, memory coalescing and reduction of warp divergence in CUDA/OpenCL programs.
4. Performance Profiling and Tuning Tools: Exploration of HPC profiling tools such as Intel VTune, NVIDIA Nsight, TAU and HPC Toolkit for performance optimization.
5. Energy-Efficient and Green High Performance Computing: Concepts of power-aware scheduling, DVFS and energy-performance trade-offs in large-scale HPC systems.

Activity Based Learning:

1. Group-based implementation of parallel programs using OpenMP/MPI and comparison of sequential vs parallel performance.
2. Case study discussions on HPC architectures and scalability analysis of real-world applications.
3. Mini project on GPU/Hybrid programming to analyze speedup, efficiency and performance optimization.

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VII

Course Title	DATA ENGINEERING		
Course Code	BCS702	CIE MARKS	50
(L : T : P) + SL	(3:0:2)+(45 Hours/Sem)	SEE MARKS	50
Credits	4	EXAM HOURS	3

COURSE LEARNING OBJECTIVES:

The purpose of the course is to facilitate the learner to:

1. Introduce the concept of Data Analytics Lifecycle.
2. Develop mathematical concepts required for advanced regression.
3. Understand data modeling in time series and its process.
4. Create awareness about text analytics and its applications.
5. Provide an overview of data analytics and visualization using Python.

Module-1	9 Hours
Introduction to Data Analytics & Lifecycle Data Analytics Lifecycle overview: Key Roles for a Successful Analytics, Background and Overview of Data Analytics Lifecycle Project. Phase 1: Discovery: Learning the Business Domain, Resources, Framing the Problem, Identifying Key Stakeholders, Interviewing the Analytics Sponsor, Developing Initial Hypotheses, Identifying Potential Data Sources. Phase 2: Data Preparation: Preparing the Analytic Sandbox, Performing ETLT, Learning About the Data, Data Conditioning, Survey and Visualize, Common Tools for the Data Preparation Phase. Phase 3: Model Planning: Data Exploration and Variable Selection, Model Selection, Common Tools for the Model Planning Phase. Phase 4: Model Building: Common Tools for the Model Building Phase. Phase 5: Communicate Results. Phase 6: Operationalize. Textbook 1: Chapter 2 (2.1 - 2.7)	
Module-2	9 Hours
Regression Models Simple Linear Regression: The Regression Equation, Fitted Value and Residuals, Least Squares.	

Multiple Linear Regression: Assessing the Model, Cross-Validation, Model Selection and Stepwise Regression, Prediction Using Regression.

Logistic Regression: Logistic Response Function and Logit, Logistic Regression and GLM, Generalized Linear Model, Predicted Values from Logistic Regression, Interpreting the Coefficients and Odds Ratios, Similarities and Differences with Linear Regression, Assessing the Models.

Textbook 1: Chapter 6 (6.1 - 6.4)

Module-3

9 Hours

Time Series

Overview of Time Series Analysis, Box-Jenkins Methodology, ARIMA Model, Autocorrelation Function (ACF), Autoregressive Models, Moving Average Models, ARMA and ARIMA Models, Building and Evaluating an ARIMA Model, Reasons to Choose and Cautions.

Textbook 1: Chapter 8 (8.1, 8.2)

Module-4

9 Hours

Text Analytics

History of Text Mining, Roots of Text Mining, Overview of Seven Practices of Text Analytics.

Applications and Use Cases: Extracting Meaning from Unstructured Text, Summarizing Text.

Text Analysis Steps, A Text Analysis Example, Collecting Raw Text, Representing Text, Term Frequency–Inverse Document Frequency (TF-IDF), Categorizing Documents by Topics, Determining Sentiments, Gaining Insights.

Textbook 1: Chapter 9 (9.1 - 9.8)

Module-5

9 Hours

Advanced Analytics – Technology and Tools

MapReduce and Hadoop, Analytics for Unstructured Data: Use Cases.

The Hadoop Ecosystem: Pig, Hive, HBase, Mahout, NoSQL.

Textbook 1: Chapter 10 (10.1 - 10.3)

PRACTICAL COMPONENT OF IPCC

Program 1: Environment Setup & Data Exploration

Problem: Install Python, Anaconda/Jupyter Notebook. Load a sample dataset (CSV).

- (i) Display first and last 10 rows.
- (ii) Identify data types.
- (iii) Find number of rows and columns.
- (iv) Generate summary statistics.

Tools: Pandas, Jupyter.

Program 2: Data Cleaning & Preprocessing

Problem: Given a dataset with missing values:

- (i) Identify null values.
- (ii) Handle missing data using mean, median and mode.
- (iii) Remove duplicate rows.
- (iv) Convert categorical columns to numeric.
- (v) Normalize numeric features.

Outcome: Clean dataset ready for analysis.

Program 3: Descriptive Statistics Analysis

Problem: For a given dataset:

- (i) Compute mean, median and mode.
- (ii) Find variance and standard deviation.
- (iii) Generate correlation matrix.
- (iv) Interpret results.

Visualization: Heatmap of correlation matrix.

Program 4: Basic Data Visualization

Problem: Using Matplotlib/Seaborn, create:

- (i) Bar chart.
- (ii) Line chart.
- (iii) Histogram.
- (iv) Scatter plot.
- (v) Box plot.

Dataset: Student marks or sales dataset.

Program 5: Advanced Visualization

Problem:

- (i) Pair plot for multi-feature dataset.
- (ii) Heatmap visualization.
- (iii) Distribution plot.
- (iv) Violin plot.

Objective: Analyze feature relationships visually.

Program 6: Exploratory Data Analysis (EDA)

Problem: Perform full EDA on a real dataset:

- (i) Outlier detection.
- (ii) Feature distribution.
- (iii) Trend analysis.
- (iv) Visualization summary.

Deliverable: EDA report with charts.

Program 7: Grouping & Aggregation

Problem: Using Pandas:

- (i) Apply groupby().
- (ii) Generate pivot tables.
- (iii) Calculate aggregated statistics.
- (iv) Visualize grouped data.

Example: Sales by region and year.

Program 8: Predictive Analytics Visualization

Problem:

- (i) Train a regression/classification model.
- (ii) Predict output.
- (iii) Visualize actual vs. predicted values.
- (iv) Plot error metrics.

Libraries: Scikit-learn, Matplotlib.

Program 9: Dashboard Creation

Problem: Using Tableau/Power BI:

- (i) Import dataset.
- (ii) Create at least four visual charts.
- (iii) Apply filters and drill-downs.
- (iv) Build an interactive dashboard.

Deliverable: Dashboard file.

Program 10: Mini Project

Problem: Choose any real-world dataset (Kaggle/Government Data).

- (i) Data cleaning.
- (ii) Exploratory Data Analysis (EDA).
- (iii) Visualization.
- (iv) Dashboard creation.
- (v) Insights report.

Examples: COVID Data Analysis, Stock Market Trends, Student Performance Analysis.

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the fundamental concepts of data analytics and visualization.
CO2	Apply regression models to perform prediction on datasets.
CO3	Analyze time series data using appropriate time series models.
CO4	Analyze text data to derive meaningful insights.
CO5	Apply analytics techniques and visualization methods using Python.

ASSESSMENT AND EVALUATION REGULATIONS OF IPCC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	20	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 20 marks.
Continuous and Comprehensive Evaluation (B)	10	-	Any one assessment method as per 24RNBE4.2 scaled to 10 marks.
Total CIE Theory (A+B)	30	12	Marks of tests and assignments awarded for a maximum of 30.

CIE Practical	12	-	Marks awarded for weekly conduction of experiments and submission of laboratory records, scaled to 12 marks.
CIE Practical Test	8	-	Average of marks of two tests, each conducted for 100 marks covering all experiments, scaled to 8.
Total CIE Practical (C)	20	8	Marks of experiments, record and test awarded for a maximum of 20 marks.
CIE: Theory + Practical (A+B+C)	50	20	CIE evaluation is done for a maximum of 50 marks.
SEE (D)	50	18	SEE is a theory exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall be 3 hours.
CIE+SEE (A+B+C+D)	100	40	Final marks shall be the sum of marks scored in components A, B, C and D.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE. The laboratory component of the IPCC shall be for CIE only. However, in SEE, the questions from the laboratory component shall be included in their respective modules only.			

Textbooks:

1. EMC Education Services, Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data, 1st Edition, John Wiley & Sons, Inc., 2015.
2. Bharti Motwani, Data Analytics using Python, 1st Edition, Wiley India Pvt. Ltd., 2020.

Reference Books:

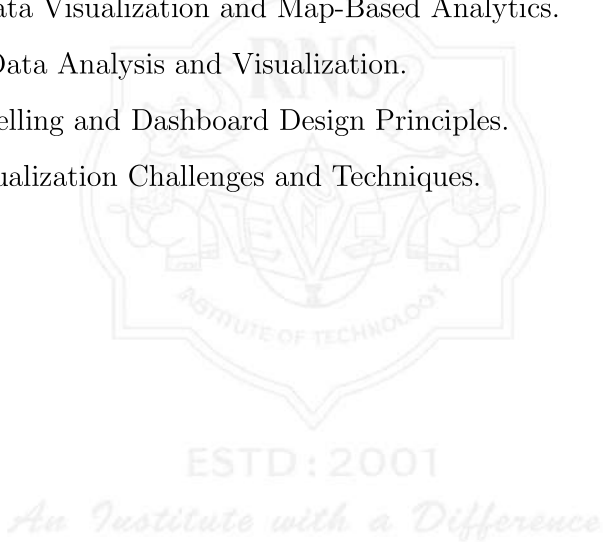
1. Tamara Munzner, Visualization Analysis and Design, 1st Edition, CRC Press (Taylor & Francis Group), 2014.
2. Cole Nussbaumer Knafllic, Storytelling with Data: A Data Visualization Guide for Business Professionals, 1st Edition, John Wiley & Sons, Inc., 2015.
3. Ryan Sleeper, Practical Tableau: 100 Tips, Tutorials and Strategies from a Tableau Zen Master, 1st Edition, O'Reilly Media, Inc., 2018.

Weblinks:

1. <https://www.coursera.org/learn/python-for-data-visualization>
2. <https://www.tableau.com/learn/training>
3. <https://pandas.pydata.org/docs/>
4. <https://matplotlib.org/stable/tutorials/index.html>

Self Study topics:

1. Interactive Data Visualization using Plotly and Dash.
2. Geospatial Data Visualization and Map-Based Analytics.
3. Time-Series Data Analysis and Visualization.
4. Visual Storytelling and Dashboard Design Principles.
5. Big Data Visualization Challenges and Techniques.



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VII

Course Title	BLOCKCHAIN TECHNOLOGY		
Course Code	BCS703	CIE MARKS	50
(L : T : P) + SL	(3:2:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	4	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand the fundamental concepts of blockchain and distributed ledger technologies.
2. Explain cryptographic foundations used in blockchain systems.
3. Analyze blockchain architectures, consensus mechanisms and platforms.
4. Design and evaluate smart contracts and decentralized applications.
5. Explore blockchain use cases, challenges and emerging trends.

Module-1	9 Hours
Introduction to Blockchain: The growth of blockchain technology, Distributed systems, The history of blockchain, Introducing Blockchain, Types of blockchain. Decentralization: Introducing Decentralization, Full ecosystem Decentralization, Decentralization in practice, Innovative trends. Textbook 1: Chapter 1, Chapter 2	
Module-2	9 Hours
Symmetric Cryptography: Introducing Cryptography, Cryptographic primitives, Advanced Encryption Standard. Asymmetric Cryptography: Asymmetric cryptography, Introducing RSA, Introducing ECC, Digital Signatures, Cryptographic constructs and blockchain technology. Textbook 1: Chapter 3, Chapter 4	
Module-3	9 Hours
Consensus Algorithms: Introducing consensus, Analysis and design, Classification, Algorithms, Choosing an Algorithm. Bitcoin Architecture: Cryptographic keys, Addresses, Transactions, Blockchain, Miners, Network wallets.	

Textbook 1: Chapter 5, Chapter 6	
Module-4	9 Hours
Bitcoins in Practice: Bitcoins in real world, Bitcoin payments, Innovation in bitcoins, Bitcoins client installation, Experimenting further with bitcoin – cli. Smart Contracts: Introducing smart contracts, Ricardian contracts, smart contract templates, oracles, Deploying smart contracts, The DAO, Advances in smart contracts. Textbook 1: Chapter 7, Chapter 8	
Module-5	9 Hours
Ethereum Architecture: Introducing Ethereum, cryptocurrency, Keys and addresses, Accounts, Transactions and Messages, Ethereum virtual machine, Blocks and blockchain, Nodes and miners, The Ethereum network, precompiled smart contracts, Wallets and client software, Supporting protocols. Ethereum in Practice: Ethereum payments, Setting up a development environment, Introducing Remix IDE, Interacting with the Ethereum blockchain with Metamask. Textbook 1: Chapter 9, Chapter 10	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain blockchain fundamentals and architectures.
CO2	Analyze cryptographic and consensus mechanisms used in blockchain systems.
CO3	Analyze blockchain platforms and smart contracts.
CO4	Apply blockchain concepts to develop decentralized applications for real-world use cases.
CO5	Analyze blockchain challenges and emerging technologies.

ASSESSMENT AND EVALUATION REGULATIONS OF IPCC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.

Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE Evaluation is done for a maximum of 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE are 20 (40%) out of allotted 50 marks to appear for SEE, with minimum 40% scored in every component of CIE.			

Textbooks:

1. Imran Bashir, Blockchain: Distributed Ledger Technology, Packt Publishing, 4th Edition, 2023.
2. Andreas M. Antonopoulos, Mastering Bitcoin: Programming the Open Blockchain, O'Reilly Media, 3rd Edition, 2023.

Reference Books:

1. Arvind Narayanan, Joseph Bonneau, Edward Felten and Andrew Miller and Steven Goldfeder, Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction, Princeton University Press, 1st Edition, 2016.
2. Melanie Swan, Blockchain: Blueprint for a New Economy, O'Reilly Media, 1st Edition, 2015.

Web Links:

1. <https://nptel.ac.in>
2. <https://www.coursera.org>
3. <https://www.edx.org>
4. <https://ocw.mit.edu>

Self Study Topics:

1. Blockchain interoperability techniques and cross-chain communication.
2. Privacy-preserving technologies: Zero-Knowledge Proofs (ZKP), zk-SNARKs.
3. Blockchain security attacks: 51% attack, Sybil attack, double-spending.
4. Regulatory and legal challenges of blockchain adoption in India and globally.
5. Blockchain applications in governance, e-voting and digital identity.
6. Emerging trends: Web3, Decentralized Finance (DeFi), Non-Fungible Tokens (NFTs).



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VII

Course Title	EXPLAINABLE SOFT COMPUTING		
Course Code	BCS714A	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The course should enable the students to:

1. To explain the principles of classical (crisp) sets, fuzzy sets and fuzzy relations, including their operations and properties.
2. To develop fuzzy system components such as membership functions, fuzzification–defuzzification methods and learning techniques.
3. To analyze and apply fuzzy logic concepts in pattern recognition, fuzzy arithmetic and fuzzy control system applications.

Module-1

9 Hours

Classical Sets and Fuzzy Set: Classical Sets: Operations on Classical Sets, Properties of Classical (Crisp) Sets, Mapping of Classical Sets to Functions, Fuzzy Sets: Fuzzy Set Operations, Properties of Fuzzy Sets, Alternative Fuzzy Set Operations.

Classical Relations and Fuzzy Relations: Cartesian Product, Crisp Relations: Cardinality of Crisp Relations, Operations on Crisp Relations, Properties of Crisp Relations, Composition, Fuzzy Relations: Cardinality of Fuzzy Relations, Operations on Fuzzy Relations, Properties of Fuzzy Relations, Fuzzy Cartesian Product and Composition.

Textbook 1: Chapter 2, Chapter 3(3.1 - 3.3)

Module-2

9 Hours

Properties of Membership Functions, Fuzzification and Defuzzification: Features of the membership function, various forms, fuzzification, defuzzification to crisp sets, λ -cuts for fuzzy relations, defuzzification to scalars.

Textbook 1: Chapter 4

Module-3

9 Hours

Development of Membership Functions: Membership Value Assignments: Intuition, Inference, Rank Ordering, Neural Networks, Genetic Algorithms, Inductive Reasoning.

Automated Methods for Fuzzy Systems: Definitions, Batch Least Squares Algorithm, Recursive Least Squares Algorithm, Gradient Method, Clustering Method, Learning from Examples, Modified Learning from Examples.

Textbook 1: Chapter 6, Chapter 7

Module-4

9 Hours

Fuzzy Pattern Recognition: Feature Analysis, Partitions of the Feature Space, Single-Sample Identification, Multifeature Pattern Recognition, Image Processing.

Fuzzy Arithmetic and the Extension Principle: Extension Principle: Crisp Functions, Mapping and Relations, Functions of Fuzzy Sets – Extension Principle, Fuzzy Transform (Mapping), Practical Considerations, Fuzzy Arithmetic, Interval Analysis in Arithmetic.

Textbook 1: Chapter 11, Chapter 12(12.1 - 12.3)

Module-5

9 Hours

Fuzzy Control Systems: Control System Design Problem: Control (Decision) Surface, Assumptions in a Fuzzy Control System Design, Simple Fuzzy Logic Controllers, Examples of Fuzzy Control System Design: Aircraft Landing Control Problem, Fuzzy Engineering Process Control: Classical Feedback Control, Fuzzy Control, Fuzzy Statistical Process Control: Measurement Data – Traditional SPC, Attribute Data – Traditional SPC, Industrial Applications.

Textbook 1: Chapter 13

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain classical (crisp) sets, fuzzy sets and fuzzy relations, including their operations, properties and mappings.
CO2	Analyze membership functions, fuzzification, defuzzification techniques and λ -cuts used in fuzzy systems.
CO3	Develop membership functions using intuition, inference, neural networks, genetic algorithms and learning-based methods.
CO4	Apply fuzzy logic principles to pattern recognition, fuzzy arithmetic, the extension principle and image processing applications.
CO5	Design fuzzy control systems for engineering and industrial applications using fuzzy logic controllers.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Timothy J. Ross, Fuzzy Logic with Engineering Applications, 3rd Edition, John Wiley & Sons, Ltd., 2010.

Reference Books:

1. Simon Haykin, Neural Networks and Learning Machines, 3rd Edition, PHI Learning Private Limited, 2008.
2. J.-S. R. Jang, C.-T. Sun and E. Mizutani, Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence, 1st Edition, Prentice Hall, 1997.

Web Links:

1. <https://www.youtube.com/watch?v=WQdVi4cw3GQ>
2. https://www.youtube.com/watch?v=JBQQaK_T8M4

-
3. https://onlinecourses.nptel.ac.in/noc26_cs44/preview
 4. <https://www.mathworks.com/help/fuzzy>
 5. <https://www.sciencedirect.com/topics/engineering/fuzzy-logic>



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VII

Course Title	DEEP LEARNING		
Course Code	BCS714B	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Introduce the fundamentals of machine learning and deep learning paradigms.
2. Build conceptual understanding of neural computation, representation learning and network depth.
3. Discuss theoretical and practical challenges involved in training deep neural networks.
4. Illustrate the role of convolutional and recurrent structures in handling spatial and temporal data.

Module-1

9 Hours

Biological & Machine Vision: The Neocognitron, LeNet-5, The Traditional Machine Learning Approach, ImageNet and the ILSVRC, AlexNet, TensorFlow Playground.

Human and Machine Language: Deep Learning for Natural Language Processing: Deep Learning Networks Learn Representations Automatically, Natural Language Processing, Computational Representations of Language: One-Hot Representations of Words, Word Vectors, Word-Vector Arithmetic, word2viz, Localist Versus Distributed Representations, Elements of Natural Human Language.

Textbook 2: Chapter 1, Chapter 2

Module-2

9 Hours

Regularization for Deep Learning: Parameter Norm Penalties, Norm Penalties as Constrained Optimization, Regularization and Under-Constrained Problems, Dataset Augmentation, Noise Robustness, Semi- Supervised Learning, Multi-Task Learning, Early Stopping, Parameter Tying and Parameter Sharing, Sparse Representations.

Textbook 1: Chapter 7 (7.1 - 7.10)

Module-3	9 Hours
Optimization for Training Deep Models: Empirical Risk Minimization, Challenges in Neural Network Optimization, Basic Algorithms: Stochastic Gradient Descent, Parameter Initialization Strategies, Algorithms with Adaptive Learning Rates: The AdaGrad algorithm, The RMSProp algorithm, Choosing the Right Optimization Algorithm. Textbook 1: Chapter 8 (8.1 - 8.5)	
Module-4	9 Hours
Convolutional Networks: The Convolution Operation, Motivation, Pooling, Convolution and Pooling as an Infinitely Strong Prior, Variants of the Basic Convolution Function, Structured Outputs, Data Types, Efficient Convolution Algorithms, Random or Unsupervised Features. Textbook 1: Chapter 9 (9.1 - 9.9)	
Module-5	9 Hours
Sequential Modelling : Recurrent and Recursive Nets: Unfolding Computational Graphs, Recurrent Neural Network, Bidirectional RNNs, Deep Recurrent Networks, Recursive Neural Networks, The Long Short-Term Memory. Textbook 1: Chapter 10 (10.1 - 10.6, 10.10)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the fundamentals of machine learning and deep learning models.
CO2	Apply feedforward neural networks and backpropagation for learning tasks.
CO3	Analyze optimization techniques used for training deep neural networks.
CO4	Design deep learning architectures for image and sequence data.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of two Internal Assessment Tests shall be scaled down to 30 marks.

Continuous and Comprehensive Evaluation (B)	20	-	Any two assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Ian Goodfellow, Yoshua Bengio and Aaron Courville, Deep Learning, 1st Edition, MIT Press, 2016.
2. John Krohn, Grant Beyleveld and Aglaé Bassens, Deep Learning Illustrated: A Visual, Interactive Guide to Artificial Intelligence, 1st Edition, Addison-Wesley Professional, 2019.

Web Links:

1. <https://www.youtube.com/watch?v=VyWAvY2CF9c>
2. <https://www.youtube.com/watch?v=7sB052Pz0sQ>
3. https://www.youtube.com/watch?v=Mubj_fqiAv8
4. <https://www.coursera.org/learn/neural-networks-deep-learning>
5. https://onlinecourses.nptel.ac.in/noc20_cs62/preview

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VII

Course Title	COMPUTER VISION AND IMAGE PROCESSING		
Course Code	BCS714C	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES:

The objectives of the course are to facilitate the learners to:

1. Understand the fundamentals of computer vision and image processing through interactive teaching methods.
2. Visualize filtering, transforms and geometric operations using videos and animations.
3. Apply collaborative learning to image restoration, segmentation and morphological processing.
4. Analyze real-world image processing problems using HOT questions and problem-based learning.
5. Implement image processing and classification algorithms through hands-on programming exercises.

Module-1		9 Hours
Introduction: What is computer vision? A brief history. Image Formation: Geometric primitives and transformations, Photometric image formation, The digital camera. Image Processing: Point operators, Linear filtering. Textbook 1: Chapter 1 (1.1 - 1.2), Chapter 2 (2.1 - 2.3), Chapter 3 (3.1 - 3.2)		
Module-2		9 Hours
Image Processing: More neighborhood operators, Fourier transforms, Pyramids and Geometric transformations. Textbook 1: Chapter 3 (3.3 - 3.6)		
Module-3		9 Hours
Image Restoration and Reconstruction: A model of image degradation/restoration process, restoration in the presence of noise only.		

Image Segmentation: Fundamentals, Point, Line and edge detection, thresholding (Foundation & Basic global thresholding only), Segmentation by region growing.

Textbook 2: Chapter 5 (5.1 - 5.3), Chapter 10 (10.1 - 10.3, 10.4)

Module-4

9 Hours

Image and Video Rendering: Image matting and compositing: Blue screen matting, Natural image matting, Smoke, shadow and flash matting, Video matting.

Texture Analysis and Synthesis: Application: Hole filling and inpainting, Application: Non-photorealistic rendering.

Video-Based Rendering: Video-based animation, Video textures.

Textbook 1: Chapter 10 (10.4.1 - 10.4.2, 10.5), Chapter 13 (13.5.1–13.5.2)

Module-5

9 Hours

Morphological Image Processing: Preliminaries, Erosion and Dilation, Opening and closing.

Feature Extraction: Background, Boundary preprocessing (Boundary following & Chain codes only).

Image Pattern Classification: Background, Patterns and classes, Pattern classification by prototype matching (Minimum distance classifier only).

Textbook 2: Chapter 9 (9.1 - 9.3, 9.5), Chapter 11 (11.1 - 11.2.2), Chapter 12 (12.1 - 12.3.1)

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the fundamentals of computer vision, image formation models and basic image processing operations.
CO2	Apply neighborhood-based image processing techniques, including Fourier transforms, pyramids, wavelets and geometric transformations.
CO3	Analyze image restoration and segmentation techniques for degraded images.
CO4	Implement image and video rendering techniques, including matting, compositing, texture synthesis and video-based rendering.
CO5	Apply morphological operations for feature extraction and pattern classification using minimum distance classifiers.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE Evaluation is done for a maximum of 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Richard Szeliski, Computer Vision: Algorithms and Applications Texts in Computer Science, 2nd Edition, Springer, 2022.
2. Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, Digital Image Processing, Pearson, 4th Edition, 2019.

Reference Books:

1. David Forsyth and Jean Ponce, Computer Vision: A Modern Approach, 2nd Edition, Pearson, 2015.
2. Reinhard Klette, Concise Computer Vision: An Introduction into Theory and Algorithms, Springer, 2014.

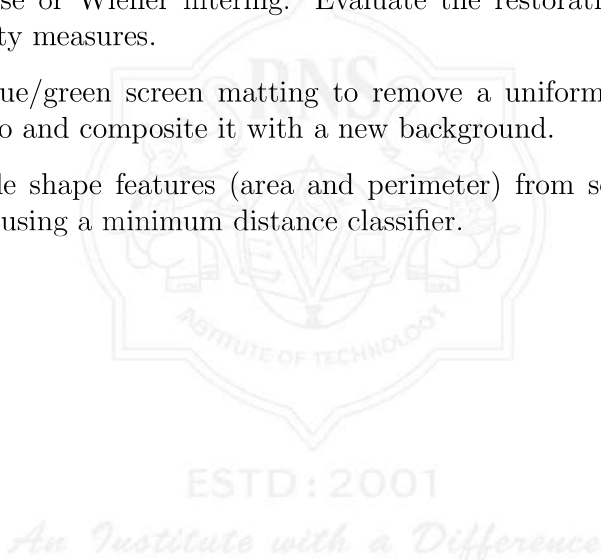
Web Links:

1. <https://youtu.be/3LaVxEX3F0o>
2. <https://youtu.be/vqRzOGg9mi8>

3. <https://youtu.be/nTILAS7N7qE>

Self Study Topics:

1. Implement a simple pinhole camera model to project 3D points onto a 2D image plane and analyse the effect of varying focal length on the projected image.
2. Perform frequency domain image filtering by computing the Fourier Transform of an image, applying a low-pass filter and reconstructing the filtered image. Compare the result with spatial domain smoothing.
3. Simulate image degradation by adding Gaussian noise to an image and restore it using inverse or Wiener filtering. Evaluate the restoration performance using suitable quality measures.
4. Implement blue/green screen matting to remove a uniform background from an image or video and composite it with a new background.
5. Extract simple shape features (area and perimeter) from segmented objects and classify them using a minimum distance classifier.



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VII

Course Title	NATURAL LANGUAGE PROCESSING AND GEN AI		
Course Code	BCS714D	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES:

The objectives of the course are to facilitate the learners to

1. Master text cleaning and tokenization techniques to resolve linguistic ambiguities and calculate string similarities.
2. Build probabilistic models and word vectors to measure how words relate and predict language sequences.
3. Understand how Transformer attention and Large Language Models generate text through bidirectional and parallel processing.
4. Learn the foundations of generative AI by implementing Autoencoders and Variational AutoEncoders to create new data.
5. Analyze advanced GPT architectures and other modern Transformers to implement sophisticated generative systems.

Module-1	9 Hours
Introduction: Knowledge in Speech and Language Processing, Ambiguity, Model and Algorithms, Language-Thought and Understanding, Some Brief History. Words and Tokens: Words, Morphemes-Parts of Words, Subword Tokenization, Regular Expressions, Rule-based tokenization, Minimum Edit Distance. Textbook 1: Chapter 1 (1.1 - 1.4, 1.6), Chapter 2 (2.1, 2.2, 2.4, 2.6, 2.8, 2.9)	
Module-2	9 Hours
N-gram Language Models: N-Grams, Evaluating Language Models: Training and Test Sets, Perplexity, Sampling sentences from a language model, Generalizing vs. overfitting the training set, Smoothing, Interpolation and Backoff. Embeddings: Lexical Semantics, Vector Semantics, Simple count-based embeddings, Cosine similarity, Word2Vec. Textbook 1: Chapter 3 (3.1 - 3.6), Chapter 5 (5.1 - 5.6)	

Module-3	9 Hours
Large Language Models: Three architectures for language models, Conditional Generation of Text: The Intuition, Prompting, Generation and Sampling.	
The Transformer: Attention, Transformer Blocks, Parallelizing computation using a single matrix, Input embeddings for token and position.	
Masked Language Models: Bidirectional Transformer Encoders, Training Bidirectional Encoders.	
Textbook 1: Chapter 7 (7.1 - 7.4), Chapter 8 (8.1 - 8.4), Chapter 9 (9.1 - 9.2)	
Module-4	9 Hours
Generative Modeling: What is Generative Modeling?, Our First Generative Model, Variational Autoencoders: Autoencoders.	
Textbook 2: Chapter 1 (1.1, 1.2), Chapter 3 (3.3)	
Module-5	9 Hours
Transformers: GPT, Analysis of GPT, Other Transformers.	
Textbook 2: Chapter 9 (9.1 - 9.3)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain the core stages of Natural Language Processing, its evolution, the role of knowledge in language understanding and linguistic ambiguity.
CO2	Implement text preprocessing techniques such as regular expressions, subword tokenization and minimum edit distance.
CO3	Apply N-gram models and vector semantics to analyze language data.
CO4	Analyze Transformer blocks, attention mechanisms and prompting techniques for text generation.
CO5	Develop fundamental generative models, including Variational Autoencoders (VAEs) and GPT-style Transformers, for text or image applications.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE Evaluation is done for a maximum of 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE+SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.

The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.

Textbooks:

1. Daniel Jurafsky and James H. Martin, Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition, 3rd Edition, Pearson Education India, 2024.
2. David Foster, Generative Deep Learning: Teaching Machines to Paint, Write, Compose and Play, 2nd Edition, O'Reilly Media, 2023.

Reference Books:

1. Tanveer Siddiqui and U. S. Tiwary, Natural Language Processing and Information Retrieval, 1st Edition, Oxford University Press, 2008.
2. James Allen, Natural Language Understanding, 2nd Edition, Benjamin-Cummings Publishing Co., 1995.
3. Raghav Bali, Generative AI with Python and TensorFlow 2: Create images, text and music with VAEs, GANs, LSTMs and GPT models, 1st Edition, Packt Publishing, 2021.

Web Links:

1. <https://nptel.ac.in>
2. <https://swayam.gov.in>

Self-Learning Activities:

1. Comparative Analysis of Transformer Architectures: Investigating the architectural trade-offs between encoder-only (BERT), decoder-only (GPT) and encoder-decoder (T5) systems regarding their efficacy in NLU tasks versus generative modeling.
2. NLP Pipeline Development and Simulation: Developing hands-on proficiency in using frameworks like Hugging Face, NLTK, or spaCy to implement text preprocessing, tokenization and sequence labeling (POS tagging, NER) on real-world datasets.
3. Vector Semantics vs. Contextual Embeddings: Evaluating how traditional models like TF-IDF and Word2Vec compare to modern self-attention mechanisms in capturing lexical similarity and long-range semantic dependencies.
4. Generative Modeling and Multi-Modal Synthesis: Analyzing the implementation of GANs (including CycleGAN), VAEs and RNNs to create multi-modal systems capable of generating diverse content across text, images and music.
5. Prompt Engineering and Few-Shot Transitions: Exploring the shift from fine-tuning small-scale models to leveraging Large Language Models (LLMs) through prompt programming and few-shot learning for task-specific optimization.
6. Generative AI Ethics and Challenges : Investigating the ethical landscape of synthetic content creation, focusing on issues of data bias, intellectual property and the potential for deepfakes, while addressing technical challenges like model hallucination and safety alignment.

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VII

Course Title	CLOUD SECURITY AND ZERO TRUST ARCHITECTURE		
Course Code	BCS714E	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The course should enable the students to:

1. Understand cloud computing concepts, including service models, deployment models, underlying technologies and cloud security standards and frameworks.
2. Explain cloud security and privacy issues, including security goals, threat models and various attack mechanisms in cloud environments.
3. Apply fundamental concepts of intrusion detection systems and security monitoring techniques in cloud infrastructures.
4. Analyze cloud security tools, attack techniques and virtualization security mechanisms such as virtual machine and hypervisor introspection.
5. Evaluate Zero Trust security principles and architectures for securing cloud-based systems.

Module-1	9 Hours
Introduction to Cloud Computing: Introduction, History and Underlying Technologies, Definitions and Characteristics, Cloud service models, Cloud deployment models, Cloud service platforms. Introduction to Cloud Security: Introduction, cloud security concepts, cloud security standards, CSA cloud reference model, NIST cloud reference model. Textbook 1: Chapter 1, Chapter 2	
Module-2	9 Hours
Cloud Security and Privacy Issues: Introduction, Cloud Security Goals/Concepts, Cloud Security Issues, Security Requirements for Privacy and Privacy Issues in cloud. Threat Model, Attacks, Defense Systems and Security Techniques: Introduction, Threat Model, Taxonomy of Attacks, Case Study. Textbook 1: Chapter 3, Chapter 4	

Module-3	9 Hours
Classification of Intrusion Detection Systems in Cloud: Introduction, TVM-based Intrusion Detection System, Hypervisor-based Intrusion Detection System, Network-based Intrusion Detection System, Distributed Intrusion Detection System, Research Challenges.	
Intrusion Detection Techniques in Cloud: Introduction, Taxonomy of IDS Techniques.	
Textbook 1: Chapter 5, Chapter 6	
Module-4	9 Hours
Overview of Tools in Cloud: Introduction, Attack Tools, Security Tools, Case Study of LibVMI – A Virtualization Specific Tool.	
Virtual Machine Introspection and Hypervisor Introspection: Virtual Machine Introspection, Hypervisor Introspection.	
Textbook 1: Chapter 7, Chapter 8	
Module-5	9 Hours
What is Zero Trust?: History and Evolution, Our Perspective on Zero Trust, Zero Trust Platform Requirements.	
Zero Trust Architectures: A Representative Enterprise Architecture, A Zero Trust Architecture, Zero Trust Deployment Models.	
Textbook 2: Chapter 2, Chapter 3	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain cloud computing fundamentals, service and deployment models, cloud security standards and security challenges in cloud environments.
CO2	Analyze cloud security and privacy issues, threat models and attacks in cloud environments.
CO3	Apply intrusion detection techniques and security monitoring methods to identify threats in cloud infrastructure.
CO4	Implement cloud security tools and virtualization security techniques such as virtual machine introspection and hypervisor introspection.
CO5	Design secure cloud systems using Zero Trust principles, authentication, segmentation and continuous monitoring.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE Evaluation is done for a maximum of 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Preeti Mishra, Emmanuel S. Pilli, R. C. Joshi, Cloud Security: Attacks, Techniques, Tools and Challenges, CRC Press, 1st Edition, 2022.
2. Jason Garbis, Jerry W. Chapman, Zero Trust Security: An Enterprise Guide, Apress Publishers, 1st Edition, 2021.

Reference Books:

1. Chris Dotson, Practical Cloud Security, O'Reilly Media, 2019.
2. Neha Agrawal, Rohit Kumar, Shashikala Tapaswi, Cloud Computing Security: Strategies and Best Practices, CRC Press, 1st Edition, 2025.
3. Tim Mather, Subra Kumaraswamy, Shahed Latif, Cloud Security and Privacy, O'Reilly Media, 2009.

Web Links

1. <https://youtu.be/WQdVi4cw3GQ>
2. https://youtu.be/JBQQaK_T8M4
3. <https://owasp.org/www-project-cloud-security/>
4. <https://csrc.nist.gov/publications/detail/sp/800-207/final>
5. <https://cloudsecurityalliance.org/research/guidance/>



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VII

Course Title	NETWORK DESIGN FUNDAMENTALS		
Course Code	BCS714F	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand fundamental principles and methodologies of network design.
2. Analyse business and technical requirements for designing enterprise networks.
3. Design campus, WAN and data center networks using industry best practices.
4. Apply security, high availability and business continuity concepts in network design.
5. Develop professional network design proposals and migration strategies.

Module-1

9 Hours

Introduction: Overview of Analysis, Architecture and Design Processes; Tactical and Strategic Significance; Importance of Network Analysis.

Systems Methodology: System Description; Service Description; Service Characteristics; Performance Characteristics.

Analysing Business Goals and Constraints: Using a Top-Down Network Design; PDIOO Network Life Cycle; Analysing Business Goals.

Textbook 1: Chapter 1 (1.1, 1.4.2, 1.4.4, 1.5 - 1.8, 1.9)

Textbook 2: Chapter 1

Module-2

9 Hours

Requirements Analysis: User Requirements; Application Requirements; Device Requirements; Network Requirements.

Developing Requirements: Developing RMA Requirements; Developing Delay and Capacity Requirements.

Characterizing Network Traffic: Identifying Major Traffic Sources & Stores; Traffic Flow Types – Client/Server, Peer-to-Peer, Distributed Computing; Characterizing Traffic Load; QoS Requirements.

Textbook 1: Chapter 2 (2.3 - 2.6), Chapter 3 (3.5 - 3.7)

Textbook 2: Chapter 4	
Module-3	9 Hours
Network Architecture: Components Architecture, Reference Architecture, Architectural Models, System and Network Architecture.	
Switching and Routing Protocols: Spanning Tree Protocol (STP, RSTP); VLAN Trunking; IP Routing – RIP, OSPF, EIGRP, BGP.	
Textbook 1: Chapter 5 (5.3 - 5.6)	
Textbook 2: Chapter 7	
Module-4	9 Hours
Security and Privacy Architecture: Developing a Security and Privacy Plan, Security and Privacy Administration, Security and Privacy Mechanisms.	
Selecting Technologies and Devices for Campus Networks: LAN Technologies and Ethernet Switching; Wireless LAN Design; Campus Cabling Topologies; Service-Level Agreements.	
Textbook 1: Chapter 9 (9.3 - 9.6); Textbook 2: Chapter 10	
Module-5	9 Hours
Network Design: Design Concept, Design Process, Network Layout, Design Traceability, Design Metrics.	
Optimizing Your Network Design: Cisco IOS Features for Optimizing Network Performance; IP Multicast Addressing.	
Textbook 1: Chapter 10 (10.2 - 10.3, 10.5 - 10.7); Textbook 2: Chapter 13	

COURSE OUTCOME

After successfully completing the course, the student will be able to:

CO1	Explain the principles, methodologies and frameworks of network design and the translation of business requirements into network architecture.
CO2	Analyze customer and organizational requirements to define the scope, constraints and design proposals for enterprise networks.
CO3	Design secure, scalable and redundant campus network architectures using layered design principles and best practices.
CO4	Design WAN and data center networks using appropriate technologies, spine-leaf topologies and virtualization strategies.

CO5	Implement network resilience strategies, QoS/CoS mechanisms and network management and automation frameworks for enterprise deployments.
-----	--

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. James D. McCabe, Network Analysis, Architecture and Design, 3rd Edition, Morgan Kaufmann Publishers, 2007.
2. Priscilla Oppenheimer, Top-Down Network Design, 3rd Edition, Cisco Press, 2011.

Reference Books

1. William Stallings, Data and Computer Communications, 10th Edition, Pearson Education, 2014.
2. Andrew S. Tanenbaum and David J. Wetherall, Computer Networks, 5th Edition, Pearson Education, 2011.

Web Links:

1. <https://www.cisco.com/c/en/us/support/docs/ip/index.html>
2. <https://www.networkworld.com/article/network-design>
3. <https://www.coursera.org/courses?query=network+design>
4. <https://youtu.be/qiQR5rTSshw>
5. https://youtu.be/3b_TAYtzuho

Self Study Topics:

1. Software-Defined Networking (SDN): Study the architecture of SDN, role of controllers and how programmable networks differ from traditional network designs.
2. IPv6 Addressing and Transition Mechanisms: Understand IPv6 address planning, subnetting strategies and dual-stack or tunneling transition techniques for enterprise networks.
3. Network Automation with Python: Explore libraries like Netmiko, NAPALM and Nornir for automated network configuration, monitoring and compliance checking.

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VII

Course Title	UI/UX DESIGN FUNDAMENTALS		
Course Code	BCS714G	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The objectives of the course are to facilitate the learners to:

1. Understand user experience design requirements, with design goals, metrics and targets.
2. Explore different prototyping methods, UX design principles with case examples.
3. Understand the role of design thinking concepts and mental models in UX design.

Module-1	9 Hours
Introduction: Usability to user experience, Emotional impact as part of user experience, User experience needs a business case, Roots of usability. Extracting Interaction Design Requirements: Introduction, Needs & Requirements, Formal requirement extraction, Methods for requirement extraction. Textbook 1: Chapter 1 (1.3 - 1.6), Chapter 5 (5.1 - 5.4)	
Module-2	9 Hours
Design Thinking, Ideation and Sketching: Introduction, Design Thinking, Design Perspectives, User Personas, Ideation, Sketching. Mental Models and Conceptual Design: Introduction, Mental models, Conceptual design, Storyboards, Design influencing user behaviour, Design for embodied interaction, Ubiquitous and situated interaction. Textbook 1: Chapter 7 (7.1 - 7.7), Chapter 8 (8.1 - 8.7)	
Module-3	9 Hours
Design Production: Introduction, Macro view of lifecycle iterations for design, Intermediate design, Detailed design, Wireframes. UX Goals, Metrics and Targets: Introduction, UX Goals, UX target tables, Work roles, user classes and UX goals, UX Measures, Measurement instruments, UX Metrics. Textbook 1: Chapter 9 (9.1 - 9.5), Chapter 10 (10.1 - 10.7)	

Module-4	9 Hours
Prototyping: Introduction, Depth & breadth of a prototype, Fidelity of prototypes, Paper prototypes, Interactivity of prototypes, Choosing the right breadth, depth, level of fidelity and amount of interactivity.	
Connections with Software Engineering: Introduction, Locus of influence in an organization, Which scenario is right for you?, Foundations for success in SE-UX development, The challenge of connecting SE and UX.	
Textbook 1: Chapter 11 (11.1 - 11.6), Chapter 23 (23.1 - 23.5)	
Module-5	9 Hours
The Interaction Cycle and the User Action Framework: Introduction, The interaction cycle, The user action framework—adding a structured knowledge base to the interaction cycle, Interaction cycle and user action framework content categories, Role of affordances within the UAF, Practical value of UAF.	
UX Design Guidelines: Introduction, Using and interpreting design guidelines, Human memory limitations, UX design guidelines & examples, Planning, Translation, Physical action, Outcomes, Assessment, Overall.	
Textbook 1: Chapter 21 (21.1 - 21.6), Chapter 22 (22.1 - 22.10)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	User experience (UX) design requirements and their role in developing user-centered systems.
CO2	Analyze design thinking concepts and mental models in UX design processes.
CO3	Apply UX design principles to achieve defined goals, metrics and performance targets.
CO4	Demonstrate prototyping techniques and their integration with software engineering practices.
CO5	Apply UX design principles through relevant case studies and examples.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Rex Hartson and Pardha S. Pyla, The UX Book: Process and Guidelines for Ensuring a Quality User Experience, 1st Edition, Morgan Kaufmann, 2012.

Web Links:

1. <https://l1nq.com/r2pruwo>
2. <https://www.edureka.co/blog/ui-ux-design-tutorial/>
3. <https://www.udemy.com/course/introtoux/>

Activity Based Learning:

1. UI design demonstrations covering UX principles using tools like Lunacy, Framer, Penpot, Visily, etc.

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VII

Course Title	CLOUD PROFESSIONAL MACHINE LEARNING ENGINEER		
Course Code	BCS714H	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The objectives of the course are to facilitate the learners to

1. Understand fundamentals of Machine Learning and Google Cloud ecosystem.
2. Learn how to design and implement ML solutions using Google Cloud tools.
3. Gain knowledge of data preparation, feature engineering and model training.
4. Apply ML models using scalable cloud infrastructure.
5. Deploy, monitor and optimize ML models in production environments.

Module-1

9 Hours

Framing ML Problems: Translating business use cases, Machine Learning approaches, ML success metrics, Responsible AI practices.

Exploring Data and Building Data Pipelines: Visualization, Statistics fundamentals, Data quality and reliability, Establishing data constraints, Running TFDV on Google Cloud Platform, Organizing and optimizing training datasets, Handling missing data, Data leakage.

Textbook 1: Chapter 1, Chapter 2

Module-2

9 Hours

Feature Engineering: Consistent data preprocessing, Encoding structured data types, Class imbalance, Feature crosses, TensorFlow Transform data and ETL tools.

Choosing the Right ML Infrastructure: Pretrained vs. AutoML vs. custom models, Pretrained models, AutoML, Custom training, Provisioning for predictions.

Architecting ML Solutions: Designing reliable, scalable and highly available ML solutions, Choosing appropriate ML service, Data collection and data management, Automation and orchestration, Serving.

Textbook 1: Chapter 3, Chapter 4, Chapter 5

Module-3	9 Hours
Building Secure ML Pipelines: Building secure ML systems, Identity and Access Management, Privacy implications of data usage and collection. Model Building: Choice of framework and model parallelism, Modelling techniques, Transfer learning, Semi-supervised learning, Data augmentation, Model generalization, Strategies to handle overfitting and underfitting. Model Training and Hyperparameters: Ingestion of various file types into training, Developing models in Vertex AI Workbench, Training a model as a job in different environments, Hyperparameter tuning, Tracking metrics during training, Retraining/redeployment evaluation, Unit testing for model training and serving. Model Explainability on Vertex AI: Model explainability techniques using Vertex AI. Textbook 1: Chapter 6, Chapter 7, Chapter 8, Chapter 9	
Module-4	9 Hours
Designing ML Training Pipelines: Orchestration frameworks, Identification of components, parameters, triggers and compute needs, System design with Kubeflow/TFX, Hybrid or multicloud strategies. Model Monitoring, Tracking and Auditing Metadata: Model monitoring, Model monitoring on Vertex AI, Logging strategy, Model and dataset lineage, Vertex AI Experiments, Vertex AI Debugging. Textbook 1: Chapter 11, Chapter 12	
Module-5	9 Hours
Maintaining ML Solutions: MLOps maturity, Retraining and versioning models, Feature store, Vertex AI permissions model, Common training and serving errors. BigQuery ML: BigQuery data access, ML algorithms, Explainability in BigQuery ML, BigQuery ML vs. Vertex AI, BigQuery design patterns. Training and Deploying TensorFlow Models at Scale: Serving TensorFlow models, Deploying a model to mobile or embedded devices, Using GPUs to speed up computation, Training models across multiple devices. Textbook 1: Chapter 13, Chapter 14 Textbook 2: Chapter 19	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Explain fundamental Machine Learning concepts, their types and the architecture and services of Google Cloud for ML solutions.
CO2	Apply data preprocessing and feature engineering techniques to improve model performance.
CO3	Build Machine Learning models using cloud-based tools such as Vertex AI, TensorFlow and Scikit-learn.
CO4	Deploy Machine Learning models in real-time and batch environments using cloud infrastructure.
CO5	Analyze model performance to optimize Machine Learning systems.

ASSESSMENT AND EVALUATION REGULATIONS OF PEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE Evaluation is done for a maximum of 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.

The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.

Textbooks:

1. Mona Ramamurthy and Pratap Ramamurthy, Official Google Cloud Certified Professional Machine Learning Engineer Study Guide, 1st Edition, John Wiley & Sons,

Inc., 2023.

2. Aurelien Geron, Hands-On Machine Learning with Scikit-Learn, Keras and Tensor-Flow: Concepts, Tools and Techniques to Build Intelligent Systems, 2nd Edition, O'Reilly Media, 2019.

Reference Books:

1. Logan Song, Journey to Become a Google Cloud Machine Learning Engineer: Build, train and deploy machine learning models at scale with Vertex AI, 1st Edition, Packt Publishing, 2022.
2. Marc Cohen, Google Cloud Platform for Machine Learning: Essential Guidance for Practitioners, 1st Edition, O'Reilly Media, 2020.

Web Links:

1. <https://cloud.google.com/certification/machine-learning-engineer>
2. <https://cloud.google.com/vertex-ai>
3. <https://www.tensorflow.org>
4. <https://www.kaggle.com>

Self Learning Activities:

1. Build ML models using Vertex AI: Learn to use Vertex AI for end-to-end model development and training. Practice working with datasets, AutoML and custom model training. Experiment with hyperparameter tuning and model evaluation. Understand how to scale training using cloud infrastructure.
2. Perform data analysis using BigQuery: Explore large datasets using SQL queries in BigQuery. Practice data cleaning, transformation and aggregation techniques. Analyze structured data to extract meaningful insights. Integrate BigQuery with ML tools for advanced analytics.
3. Create ML pipelines using Dataflow: Develop pipelines for data ingestion and pre-processing. Work with batch and streaming data processing methods. Automate data workflows required for ML model training. Gain experience in handling large-scale distributed data.
4. Deploy ML models with REST APIs: Learn to deploy trained models as RESTful services. Practice sending requests and receiving predictions via APIs. Understand integration of models into applications. Explore deployment using cloud endpoints and containers.
5. Implement real-time prediction systems: Work on building systems that provide instant predictions. Use streaming data and deployed ML models. Apply concepts to real-world use cases like recommendations. Understand latency, scalability and performance optimization.

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VII

Course Title	INTRODUCTION TO ALGORITHMS		
Course Code	BCS755A	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand the methods of analyzing algorithms and to analyze evaluating their performance.
2. Apply algorithm's efficiencies using asymptotic notations.
3. Analyze problems using algorithm design methods such as the brute force method, greedy method, divide and conquer, decrease and conquer, transform and conquer and dynamic programming.
4. Design the appropriate data structure and algorithm design method for a specified application.

Module-1		9 Hours
Introduction: What is an Algorithm? Fundamentals of Algorithmic Problem Solving, Important problem Types, Fundamental Data Structures. Fundamentals of the Analysis of Algorithm Efficiency: Analysis Framework, Asymptotic Notations and Basic Efficiency Classes. Textbook 1: Chapter 1 (1.1 - 1.4), Chapter 2 (2.1, 2.2)		
Module-2		9 Hours
Fundamentals of the Analysis of Algorithm Efficiency: Mathematical Analysis of Non recursive Algorithms, Mathematical Analysis of Recursive Algorithms. Brute Force: Selection Sort and Bubble Sort, Sequential Search and Brute Force String Matching. Textbook 1: Chapter 2 (2.3, 2.4), Chapter 3 (3.1, 3.2)		
Module-3		9 Hours
Exhaustive Search: Travelling Salesman problem and Knapsack Problem, Depth First search and Breadth First search.		

Decrease-and-Conquer: Insertion Sort, Topological Sorting.

Divide and Conquer: Merge Sort, Quick sort, Binary Tree Traversals.

Textbook 1: Chapter 3 (3.4, 3.5), Chapter 4 (4.1, 4.2), Chapter 5 (5.1 - 5.3)

Module-4

9 Hours

The Greedy Method: Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffman Trees and Codes.

Dynamic Programming: Three basic examples, The Knapsack Problem and Memory Functions.

Textbook 1: Chapter 9 (9.1 - 9.4), Chapter 8 (8.1 - 8.2)

Module-5

9 Hours

Transform-and-Conquer: Heap and Heapsort.

Space and Time Trade-offs: Sorting by Counting: Comparison counting sort, Input Enhancement in String Matching: Horspool's Algorithm.

Textbook-2: Chapter 6 (6.4), Chapter 7 (7.1, 7.2)

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Analyze algorithm efficiency using asymptotic notations and fundamental algorithmic problem-solving concepts.
CO2	Analyze brute-force, decrease-and-conquer and divide-and-conquer algorithms for sorting, searching and matrix problems.
CO3	Design algorithms for optimization problems using greedy and dynamic programming paradigms.
CO4	Apply transform-and-conquer and space-time trade-off strategies to sorting and string matching problems.

ASSESSMENT AND EVALUATION REGULATIONS OF OEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of two Internal Assessment Tests shall be scaled down to 30 marks.

Continuous and Comprehensive Evaluation (B)	20	-	Any two assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE evaluation maximum 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Anany Levitin, Introduction to the Design and Analysis of Algorithms, 3rd Edition, Pearson Education, 2012.

Reference Books:

1. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, Computer Algorithms/C++, 2nd Edition, Universities Press, 2008.
2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, Introduction to Algorithms, 3rd Edition, MIT Press, 2009.
3. A. M. Padma Reddy, Design and Analysis of Algorithms, 1st Edition, Sri Nandi Publications, 2009.

Web Links:

1. <https://youtu.be/xGYsEqe9Vl0>
2. <https://youtu.be/A03oI0znAoc>
3. <https://slink.com/64ueq97>
4. <https://www.udemy.com/topic/algorithms/>

Self Learning:

1. Identify algorithms used in daily life (ATM, Google Maps).

-
2. Compare two sorting algorithms based on time complexity.
 3. Write pseudocode for a simple problem.
 4. Trace Fibonacci / searching algorithms manually.
 5. Explore algorithm visualization tools online.



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VII

Course Title	FOUNDATIONS OF INFORMATION SECURITY		
Course Code	BCY755B	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. To introduce fundamental principles of information security and its importance in modern systems.
2. To provide knowledge on identification and authentication mechanisms used for user verification.
3. To familiarize access control concepts and models for regulating resource access.
4. To introduce auditing and accountability mechanisms for ensuring system security and traceability.
5. To present the fundamentals of cryptography and its role in securing data across different states.

Module-1	9 Hours
What is Information Security? Defining Information Security, When Are You Secure? Models for Discussing Security Issues, Attacks, Défense in Depth.	
Textbook: Chapter 1	
Module-2	9 Hours
Identification and Authentication: Identification, Authentication, Common Identification and Authentication methods.	
Textbook: Chapter 2	
Module-3	9 Hours
Authorization and Access Controls: What are Access Controls? Implementing Access Controls, Access Control Models, Physical Access Controls.	
Textbook: Chapter 3	

Module-4	9 Hours
Auditing and Accountability: Accountability, Security Benefits of Accountability, Auditing.	
Textbook: Chapter 4	
Module-5	9 Hours
Cryptography: The History of Cryptography, Modern Cryptographic Tools, Protecting Data at Rest, in Motion and in Use.	
Textbook: Chapter 5	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Apply information security concepts, attacks, and defense-in-depth.
CO2	Demonstrate authentication methods for user verification.
CO3	Analyze access control models and mechanisms for regulating resource access.
CO4	Examine auditing and accountability for ensuring security and traceability.
CO5	Implement cryptographic techniques for protecting data at rest, in transit, and in use.

ASSESSMENT AND EVALUATION REGULATIONS OF OEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	–	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Alternate Assessment (B)	20	–	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE Evaluation is done for a maximum of 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.

CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. Jason Andress, Foundations of Information Security: A Straightforward Introduction, 1st Edition, No Starch Press, 2019.

Reference Books:

1. Michael E. Whitman and Herbert J. Mattord, Principles of Information Security, 7th Edition, Cengage Learning, 2021.
2. Mark Stamp, Information Security: Principles and Practice, 1st Edition, John Wiley & Sons, Inc., 2005.

Web Links:

1. <https://nptel.ac.in/courses/106105162>
2. https://www.youtube.com/results?search_query=Cyber+Security+Full+Course+for+Beginners
3. https://www.youtube.com/results?search_query=Introduction+to+Cryptography+and+Network+Security+Neso+Academy
4. <https://iamsridhariyer.com/infosec/>

Activity Based Learning:

1. Cryptography Demonstration Activity: Demonstration of basic encryption techniques using simple tools like Online AES / RSA demo tools.
2. Network Security Demo (Visualization Based): Demonstration of: Firewall working, Packet filtering using tools such as Cisco Packet Tracer

Self Study Topics:

1. Zero Trust Security Model in modern networks
2. Cybersecurity frameworks (NIST, ISO 27001 overview)
3. Security in cloud computing environments
4. Blockchain applications in information security
5. Ethical hacking and penetration testing basics

B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VII

Course Title	FULL STACK WEB DEVELOPMENT		
Course Code	BCD755C	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The purpose of the course is to facilitate the learner to:

1. Understand the essential JavaScript concepts for web development.
2. Distinguish different types of events and their roles in interactive web applications.
3. Utilize React JS to build front end User Interface.
4. Understand the usage of API's to create web applications using Express JS.
5. Store and model data in a no sql database.

Module-1

9 Hours

Basic JavaScript Instructions: Statements, Comments, Variables, Data Types, Arrays, String operator, Functions, Methods & Objects, Decisions & Loops.

Textbook 1: Chapter 2, Chapter 3, Chapter 4

Module-2

9 Hours

Document Object Model: DOM, DOM Tree, Accessing Elements, Selecting Elements, Adding Elements using DOM Manipulation, Adding an element to the DOM Tree, Removing elements via DOM Manipulation, Removing elements from the DOM Tree, CROSS-SITE scripting attacks, Attribute nodes.

Events: Different Types of Events, How to bind an event to an element, Event Listeners, Event Delegation.

Textbook 1: Chapter 5, Chapter 6

Module-3

9 Hours

Form enhancement and validation: Helper Functions, The form element, Form controls, Submitting forms, HTML5 elements & Attributes.

Introduction: MERN, MERN components, Why MERN.

Hello World: Server- Less Hello world, Express, Separate Script File, JSX Transform, Older Browsers Support, Automate.

React Components: Issue Tracker, React Classes, Composing Components, Passing Data using Properties, Passing Data Using Children, Dynamic Composition.

Textbook 1: Chapter 13

Textbook 2: Chapter 1, Chapter 2, Chapter 3

Module-4

9 Hours

React State: Initial State, Async State Initialization, Updating State, Lifting State Up, Event Handling, Stateless Components, Designing Components.

Express and GraphQL: Express, REST API, GraphQL.

Textbook 2: Chapter 4, Chapter 5

Module-5

9 Hours

MongoDB: MongoDB Basics, MongoDB CRUD Operations, MongoDB Node.js Driver, Schema Initialization, Reading from MongoDB, Writing to MongoDB.

Modularization and Webpack: Back-End Modules, Front-End Modules and Webpack, Transform and Bundle, Libraries Bundle.

Textbook 2: Chapter 6, Chapter 8

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Understand JavaScript to build dynamic and interactive Web projects.
CO2	Implement user interface components for JavaScript-based Web using React.JS
CO3	Apply Express/Node to build web applications on the server side.
CO4	Develop data model in an open-source NoSQL database.
CO5	Demonstrate modularization and packing of the front-end modules.

ASSESSMENT AND EVALUATION REGULATIONS OF OEC

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.

Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE Evaluation is done for a maximum of 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE. If any courses of 3 credits are of Integrated Course type, for such courses the method suggested for 4 credits IPCC shall be followed.			

Textbooks:

1. Jon Duckett, JavaScript & jQuery: Interactive Front-End Web Development, 1st Edition, John Wiley & Sons, Inc., 2014.
2. Vasana Subramanian, Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, 2nd Edition, Apress, 2019.

Web Links:

1. <https://github.com/vasansr/pro-mern-stack>
2. <https://nptel.ac.in/courses/106106156>
3. <https://archive.nptel.ac.in/courses/106/105/106105084/>

Activity Based Learning:

1. Course Project: Build Web applications using MERN stack. Students (group of 2) can choose any real-world problem from domains such as finance, marketing, medical, or enterprise projects.

Self study topics:

1. Microservices architecture
2. GraphQL vs REST

-
3. WebSockets (real-time apps)
 4. Full-stack AI integration (LLM APIs)
 5. DevOps for MERN (Docker, CI/CD)



B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – VII

Course Title	INTRODUCTION TO MACHINE LEARNING		
Course Code	BCI755D	CIE MARKS	50
(L : T : P) + SL	(3:0:0)+(45 Hours/Sem)	SEE MARKS	50
Credits	3	EXAM HOURS	3

COURSE LEARNING OBJECTIVES

The objectives of the course are to facilitate the learners to:

1. Introduce the fundamental concepts and techniques of machine learning.
2. Understand the types of machine learning and the challenges faced in real world applications.
3. Familiarize the machine learning algorithms such as regression, decision trees, Bayesian models, clustering, and neural networks.
4. Explore advanced concept like reinforcement learning and provide practical insight into its applications.
5. Enable students to model and evaluate machine learning solutions for different types of problems.

Module-1	9 Hours
Introduction to Machine Learning: Need for Machine Learning, Machine Learning Explained, Machine Learning in Relation to other Fields, Types of Machine Learning, Challenges of Machine Learning, Machine Learning Process, Machine Learning Applications. Understanding Data: Introduction, Big Data Analysis Framework, Descriptive Statistics, Univariate Data Analysis and Visualization. Textbook 1: Chapter 1 (1.1 - 1.7), Chapter 2 (2.1 - 2.5)	
Module-2	9 Hours
Understanding Data: Bivariate Data and Multivariate Data, Multivariate Statistics, Essential Mathematics for Multivariate Data, Feature Engineering and Dimensionality Reduction Techniques. Basic Learning Theory: Design of Learning System, Introduction to Concept of Learning, Modelling in Machine Learning. Textbook 1: Chapter 2 (2.6 - 2.8, 2.10), Chapter 3 (3.3 - 3.4, 3.6)	

Module-3	9 Hours
Similarity-based Learning: Nearest-Neighbour Learning, Weighted K-Nearest-Neighbor Algorithm, Nearest Centroid Classifier, Locally Weighted Regression (LWR). Regression Analysis: Introduction to Regression, Introduction to Linear Regression, Multiple Linear Regression, Polynomial Regression, Logistic Regression. Decision Tree Learning: Introduction to Decision Tree Learning Model, ID3 Tree Construction. Textbook 1: Chapter 4 (4.2-4.5), Chapter 5 (5.1 - 5.3, 5.5 - 5.7), Chapter 6 (6.1, 6.2.1)	
Module-4	9 Hours
Bayesian Learning: Introduction to Probability-based Learning, Fundamentals of Bayes Theorem, Classification Using Bayes Model, Naïve Bayes Algorithm for Continuous Attributes. Artificial Neural Networks: Introduction, Biological Neurons, Artificial Neurons, Perceptron and Learning Theory, Types of Artificial Neural Networks, Popular Applications of Artificial Neural Networks, Advantages and Disadvantages of ANN, Challenges of ANN. Textbook 1: Chapter 8 (8.1 - 8.4), Chapter 10 (10.1 - 10.5, 10.9 - 10.11)	
Module-5	9 Hours
Clustering Algorithms: Introduction to Clustering Approaches, Proximity Measures, Hierarchical Clustering Algorithms, Partitional Clustering Algorithm, Density-based Methods, Grid-based Approach. Reinforcement Learning: Overview of Reinforcement Learning, Scope of Reinforcement Learning, Reinforcement Learning as Machine Learning, Components of Reinforcement Learning, Markov Decision Process, Multi-Arm Bandit Problem and Reinforcement Problem Types, Model-based Learning. Textbook 1: Chapter 13 (13.1 - 13.6), Chapter 14 (14.1 - 14.7)	

COURSE OUTCOMES

After successfully completing the course, the student will be able to:

CO1	Describe the machine learning techniques, their types and data analysis framework.
CO2	Apply mathematical concepts for feature engineering and perform dimensionality reduction to enhance model performance.
CO3	Develop similarity-based learning models and regression models for solving classification and prediction tasks.

CO4	Build probabilistic learning models and design neural network models using perceptron's and multilayer architectures
CO5	Utilize clustering algorithms to identify patterns in data and implement reinforcement learning techniques

ASSESSMENT AND EVALUATION REGULATIONS OF OEC

Assessment Type	Maximum Marks	Minimum Pass-ing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Two tests each of 50 marks shall be conducted. Average of Two Internal Assessment Tests shall be scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	-	Any two Assessment methods as per 24RNBE4.2 (*if it is project based, one assignment shall be given), scaled to 20 marks.
Total CIE (A+B)	50	20	CIE Evaluation is done for a maximum of 50 marks.
SEE (C)	50	18	SEE is a theory exam conducted for 100 marks and scaled down to 50 marks. Exam duration shall be for 3 hours.
CIE + SEE (A+B+C)	100	40	Final marks shall be the sum of marks scored in Components A, B and C.
The minimum marks to be secured in CIE is 20 (40%) out of allotted 50 marks to appear for SEE.			

Textbooks:

1. S. Sridhar and M. Vijayalakshmi, Machine Learning, 1st Edition, Oxford University Press, 2021.

Reference Books:

1. M. Narasimha Murty and V. S. Ananthanarayana, Machine Learning: Theory and Practice, 1st Edition, Universities Press, 2024.
2. Tom M. Mitchell, Machine Learning, 1st Edition, McGraw-Hill Education, 1997.
3. Andriy Burkov, The Hundred-Page Machine Learning Book, 1st Edition, Andriy Burkov, 2019.

Web Links:

1. <https://www.universitiespress.com/resources?id=9789393330697>
2. https://www.drssridhar.com/?page_id=1053
3. <https://www.geeksforgeeks.org/machine-learning/>
4. https://www.tutorialspoint.com/machine_learning/index.htm
5. https://onlinecourses.nptel.ac.in/noc22_cs29/preview





VISION

Building RNSIT into a World Class Institution.



MISSION

To impart high quality education in Engineering, Technology and Management with a difference, enabling students to excel in their career by

- Attracting quality students and preparing them with a strong foundation in fundamentals so as to achieve distinctions in various walks of life leading to outstanding contributions.
- Imparting value based, need based, choice based and skill based professional education to the aspiring youth and carving them into disciplined, World class professionals with social responsibility.
- Promoting excellence in Teaching, Research and Consultancy that galvanizes academic consciousness among Faculty and Students.
- Exposing Students to emerging frontiers of knowledge in various domains and make them suitable for Industry, Entrepreneurship, Higher studies, and Research & Development.
- Providing freedom of action and choice for all the Stakeholders with better visibility.

 www.rnsit.ac.in

follow us on:    

RR Nagar, Dr. Vishnuvardhan Road, Bengaluru - 560 098
Ph: (080) 28611880, 28611881