RNS TRUST ®

RNS INSTITUTE OF TECHNOLOGY

Autonomous Institution affiliated to Visvesvaraya Technological University, Belagavi
Approved By AICTE, New Delhi. Accredited by NAAC 'A+' Grade
Channasandra, Dr. Vishnuvardhan Road, Bengaluru - 560 098
Ph:(080)28611880,28611881 URL: www.rnsit.ac.in

ESTD.: 2001
An Institute with a Difference

Dr. R N Shetty
Founder

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

BACHELOR OF ENGINEERING

Scheme and Syllabus of III & IV Semesters

2024 SCHEME

INSTITUTION'S INNOVATION COUNCIL
(Ministry of HRD Initiative)

www.rnsit.ac.in

Scheme and Syllabus
Batch 2024 - 2028

(As per NEP-2020)
(Effective from the academic year 2024 – 25)

About RNSIT

RNS Institute of Technology, familiar as RNSIT was established in the year 2001. It is the brainchild of Dr. R. N. Shetty, a well-known industrialist, and a great philanthropist. The institution was established with the sole purpose of imparting quality education in Engineering, Technology and Management. Being one of the premier engineering colleges in the country RNSIT is marching towards placing itself in the global map by excelling in academics and placements.

Lush green campus, state of the art infrastructure, well equipped laboratories, hundreds of activities, numerous new initiatives, consistent VTU ranks, path breaking placements, cultural extravaganza, Sports events, 4000 + students and 300 + faculty – with all these RNSIT has become dream destination for engineering aspirants across the country.

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

About Computer Science and Engineering

The Department of Computer Science & Engineering (CSE) at RNSIT is where academic excellence and innovation converge. For over 24 years, the department has proudly shaped future engineers, providing quality education, cutting-edge research opportunities, and a strong foundation for successful careers. Accredited by the NBA, the department is committed to state-of-the-art labs and 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 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 various specialized fields. It has organized over 50 FDPs, 4 International Conferences, 3 National Conferences, and 400+ invited talks, benefiting 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 Global Environment.
- Educating and preparing the graduates, highly sought after, productive, and well-respected for their work culture.
- Supporting and inducing lifelong learning.

Programme Educational Objectives

A Graduate of Computer Science and Engineering will be able to:

- **PEO1**: Demonstrate their expertise in solving contemporary problems through design, analysis, implementation and evaluation of hardware and software systems.
- **PEO2**: Engage in the Computer Science and Engineering profession locally and globally by contributing ethically to the competent and professional practice of Engineering or other professional careers.
- **PEO3**: Adapt to a constantly changing world through professional development and sustained learning.
- **PEO4**: Exhibit leadership and entrepreneurship skills by incorporating organizational goals and providing facilities for peer members with defined objectives.
- **PEO5**: Develop communication skills and show a commitment to teamwork necessary to function productively and professionally on multidisciplinary teams.

Program Outcomes (Revised)

1. **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.

2. **PO2: Problem Analysis**: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

3. **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 the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

4. **PO4: Conduct Investigations of Complex Problems**: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

5. **PO5: Engineering Tool Usage**: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

6. **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).

7. **PO7: Ethics**: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)
8. **PO8: Individual and Collaborative Teamwork**: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.
9. **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, make effective presentations considering cultural, language, and learning differences
10. **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 and in multidisciplinary environments.
11. **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)

Programme Specific Outcomes (PSOs)

The Computer Science and Engineering graduates will have

- **PSO1**: Solid foundation in the principles and practices of computer science, including mathematics, science, and basic engineering for design, develop, test, and maintain Hardware/Software Systems.
- **PSO2**: Professional skills to function as members of multi-disciplinary teams and to communicate effectively using modern tools.
- **PSO3**: n ability to pursue their careers in the software industry or higher studies and continue to develop their professional knowledge in the areas like Data Mining, Networking, Image Processing, AI & Machine learning, Data Science, IOT.
- **PSO4**: An ability to practice the profession with ethics, integrity, leadership and social responsibility.

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

3rd Semester

SL NO	COURSE	COURSE CODE	COURSE TITLE	TD/ PSB	TEACHING HOURS PER WEEK				EXAMINATION				CREDITS
					THEORY LECTURE (L)	TUTORIAL (T)	PRACTICAL (P)	(TW + SL)*/Sem (S)	DURATION IN HOURS	CIE MARKS	SEE MARKS	TOTAL MARKS	
1	PCC	BCS301	DISCRETE MATHEMATICS AND STATISTICAL INFERENCE	CSE / MATHS	3	0	0	45	3	50	50	100	3
2	IPCC	BCS302	LOGIC DESIGN AND COMPUTER ORGANIZATION	CSE	3	0	2	45	3	50	50	100	4
3	IPCC	BCS303	OPERATING SYSTEMS	CSE	3	0	2	45	3	50	50	100	4
4	PCC	BCS304	DATA STRUCTURES AND APPLICATIONS	CSE	3	0	0	45	3	50	50	100	3
5	PCCL	BCSL305	DATA STRUCTURES LABORATORY	CSE	0	0	2	0	3	50	50	100	1
6	ESC	BXX306X	ESC/ETC/PLC (BUCKET LIST – A PROVIDED)	CSE	3	0	0	45	3	50	50	100	3
					2	0	2	30					
7	UHV	BSCK307	SOCIAL CONNECT AND RESPONSIBILITY	ANY	0	0	2	0	2	100	0	100	1
8	AEC/ SEC	BXXX/ L358X	BUCKET LIST – B PROVIDED	CSE	IF COURSE IS THEORY				2	50	50	100	1
					1	0	0	15					
					IF COURSE IS PRACTICAL				3				
					0	0	2	0					
9	MC	BNSK359	NATIONAL SERVICE SCHEME (NSS)	ANY	0	0	2	0		100	0	100	0
		BPEK359	PHYSICAL EDUCATION (PE) (SPORTS AND ATHLETICS)	PED									
		BYOK359	YOGA	PED									
	TOTAL									550	350	900	20

*** TW – Term Work, SL – Self Learning**

Additional Mathematics I and II:

(1) These courses are prescribed for III and IV semesters respectively to lateral entry Diploma holders admitted to III semester of B.E./B.Tech., programs. They shall attend the classes during the respective semesters to complete all the formalities of the course and appear for the Continuous Internal Evaluation (CIE). In case, any student fails to register for the said course/fails to secure the minimum 40 % of the prescribed CIE marks, he/she shall be deemed to have secured an F grade. In such a case, the student has to fulfill the course requirements during subsequent semester/s to earn the qualifying CIE marks. These courses are slated for CIE only and have no SEE.

(2) Additional Mathematics I and II shall not be considered for vertical progression as well as for the calculation of SGPA and CGPA, but completion of the courses shall be mandatory for the award of degree.

(3) Successful completion of the courses Additional Mathematics I and II shall be indicated as satisfactory in the grade card. Non-completion of the courses Additional Mathematics I and II shall be indicated as Unsatisfactory.

EMERGING SCIENCE COURSES / EMERGING TECHNOLOGY COURSES BUCKET LIST - A	
BCS306A	OBJECT ORIENTED PROGRAMMING USING JAVA
BCS306B	INTRODUCTION TO DATA SCIENCE
BCD306C	DATA ANALYTICS WITH R
BCS306D	HASKELL PROGRAMMING

ABILITY ENHANCEMENT COURSES / SKILL DEVELOPMENT COURSES BUCKET LIST - B	
BFLC358A	FOREIGN LANGUAGE COURSE 1- GERMAN
BFLC358B	FOREIGN LANGUAGE COURSE 1- SPANISH
BCIL358C	AI TOOLS AND APPLICATIONS
BCST358D	GREEN IT AND SUSTANABILITY
BCSL358E	PROJECT MANAGEMENT WITH GIT
BFLC358A	FOREIGN LANGUAGE COURSE 1- GERMAN

4th Semester

SL NO	COURSE	COURSE CODE	COURSE TITLE	TD/ PSB	TEACHING HOURS PER WEEK				EXAMINATION				CREDITS
					THEORY LECTURE (L)	TUTORIAL (T)	PRACTICAL (P)	TW + SL /Sem (S)	DURATION IN HOURS	CIE MARKS	SEE MARKS	TOTAL MARKS	
1	PCC	BCS401	DESIGN AND ANALYSIS OF ALGORITHMS	CSE	3	2	0	45	3	50	50	100	4
2	IPCC	BCS402	DATABASE MANANGEMENT SYSTEMS	CSE	3	0	2	45	3	50	50	100	4
3	IPCC	BCS403	MICROCONTROLLERS AND IOT	CSE	3	0	2	45	3	50	50	100	4
4	PCCL	BCSL404	DESIGN AND ANALYSIS OF ALGORITHMS LABORATORY	CSE	0	0	2	0	3	50	50	100	1
5	ESC	BXX405X	ESC/ETC/PLC (BUCKET LIST – C PROVIDED)	CSE / Maths	3	0	0	45	3	50	50	100	3
					2	0	2	30					
6	AEC/ SEC	BXXT/L456X	BUCKET LIST – D PROVIDED	CSE	IF COURSE IS THEORY				2	50	50	100	1
					1	0	0	15					
					IF COURSE IS PRACTICAL				3				
					0	0	2	0					
7	BSC	BENS407/ BBOE407	ENVIRONMENTAL STUDIES / BIOLOGY FOR ENGINEERS	CIV / CSE	2	0	0	30	2	50	50	100	2
8	UHV	BUHV408	UNIVERSAL HUMAN VALUES	ANY	0	0	2	0	2	50	50	100	1
9	MC	BNSK459	NATIONAL SERVICE SCHEME (NSS)	ANY	0	0	2	0	-	100	0	100	0
		BPEK459	PHYSICAL EDUCATION (PE) (SPORTS AND ATHLETICS)	PED									
		BYOK459	YOGA	PED									
	TOTAL									550	350	900	20

NOTE: (i) 1st semester Chemistry cycle sections will study "ENVIRONMENTAL STUDIES" in 4th semester and "BIOLOGY FOR ENGINEERS" in 5th semester
(ii) 1st semester Physics cycle sections will study "BIOLOGY FOR ENGINEERS" in 4th semester and "ENVIRONMENTAL STUDIES" in 5th semester

***TW – Term Work, SL – Self Learning**

EMERGING SCIENCE COURSES / EMERGING TECHNOLOGY COURSES BUCKET LIST - C	
BCS405A	LINEAR ALGEBRA
BCS405B	GRAPH THEORY
BCS405C	METRIC SPACES
BCS405D	GAME THEORY

ABILITY ENHANCEMENT COURSES / SKILL DEVELOPMENT COURSES BUCKET LIST - D	
BFAT456A	FINANCE AND ACCOUNTING
BCST456B	INTRODUCTION TO UNIX
BCSL456C	MOBILE APP DEVELOPMENT
BCSL456D	ADVANCED JAVA

III SEMESTER SUBJECTS

B.E. (CSE and Allied Branches)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – III

Course Title:	**Discrete Mathematics and Statistical Inference**		
Course Code:	**BCS301**	**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. To help students to understand discrete and continuous mathematical structures.
2. To impart basics of relations and functions.
3. To facilitate students in applying principles of Recurrence Relations to find the generating functions and solve the Recurrence relations.
4. To provide the principles of statistical inferences and the basics of hypothesis testing with emphasis on some commonly encountered hypothesis.

Module-1 **9 Hours**

The Fundamentals of Logic: Basic Connectives and Truth Tables, Logic Equivalence – The Laws of Logic, Logical Implication – Rules of Inference. The Use of Quantifiers, Quantifiers, Definitions and Proofs of Theorems.

Textbook 1: Ch - 2.1 to 2.5

Module-2 **9 Hours**

Mathematical Induction, The Well Ordering Principle – Mathematical Induction, Recursive Definitions.

Fundamental Principles of Counting: The Rules of Sum and Product, Permutations, Combinations –

The Binomial Theorem, Combinations with Repetition.

Textbook 1: Ch- 1.1 to 1.41, 4.1 to 4.21

Module-3 **9 Hours**

Cartesian Products and Relations, Functions – Plain and One-to-One, Onto Functions.

The Pigeon-hole Principle, Function Composition and Inverse Functions.

Properties of Relations, Computer Recognition – Zero-One Matrices and Directed Graphs, Partial Orders – Hasse Diagrams, Equivalence Relations and Partitions.

Textbook 1: Ch - 5.1 to 5.3, 5.6 to 5.11

Module-4 **9 Hours**

The Principle of Inclusion and Exclusion, Generalizations of the Principle,

Derangements – Nothing is in its Right Place, Rook Polynomials.

Recurrence Relations: First Order Linear Recurrence Relation, The Second Order Linear Homogeneous Recurrence Relation with Constant Coefficients.

Textbook 1: Ch - 8.1 to 8.5, 10.1 to 10.2

Module-5 **9 Hours**

Introduction to Statistical Inference, Point Estimation, Confidence Intervals, Hypothesis Testing - Basics: Null and Alternative Hypothesis, Type I and Type II Errors, z – test for large samples, t-Test (One-Sample, Two-Sample, Paired), Chi-Square Test for Independence & Goodness-of-Fit. **Textbook 1: Ch- 27.1 to 27.8**

COURSE OUTCOMES:

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

CO1	**Apply** logical reasoning and various proof strategies (direct, contradiction, induction) to validate mathematical theorems and logical statements.
CO2	**Explain and analyze** discrete structures such as sets, logic, and combinatorics to identify their role in solving computational problems.
CO3	**Apply and analyze** relations, functions, and partially ordered sets to model and interpret discrete mathematical structures in computing systems.
CO4	**Solve and evaluate** recurrence relations and generating functions in the context of algorithm analysis and problem modelling.
CO5	**Apply and interpret** statistical estimation and hypothesis testing to assess engineering problems and evaluate conclusions using confidence intervals.

ASSESSMENT DETAILS (Both CIE and SEE)			
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. **Ralph P. Grimaldi**: "Discrete and Combinatorial Mathematics", 5th Edition, Pearson Education. 2004.
2. **B. S. Grewal**: "Higher Engineering Mathematics", Khanna publishers, 44th Ed., 2021.

Reference Books

1. **Ralph P. Grimaldi, B V Ramana**: "Discrete Mathematical Structures an Applied Introduction", 5th Edition, Pearson Education, 2004.
2. **B. V. Ramana:** "Higher Engineering Mathematics" McGraw-Hill Education, 11th Ed., 2017.
3. **S C Gupta, V K Kapoor:** "Fundamentals of Mathematical Statistics", Sultan Chand & Sons Publication, 12th edition.
4. **H. K. Dass and Er. Rajnish Verma:** "Higher Engineering Mathematics" S. Chand Publication, 3rd Ed., 2014.

Web links and Video Lectures (e-Resources):

1. http://nptel.ac.in/courses.php?disciplineID=111
2. http://www.class-central.com/subject/math(MOOCs)
3. http://academicearth.org/
4. http://www.themathpage.com/
5. http://www.abstractmath.org/
6. http://www.ocw.mit.edu/courses/mathematics/

Activity-Based Learning (Suggested Activities in Class)/Practical-Based Learning

- Quizzes
- Assignments
- Seminar

B.E. (Computer Science and Engineering) Outcome Based Education (OBE) and Choice Based Credit System (CBCS) SEMESTER – III			
Course Title:	**Logic Design and Computer Organization**		
Course Code:	**BCS302**	**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. Demonstrate the functionalities of binary logic system
2. Explain the working of combinational and sequential logic system
3. Realize the basic structure of computer system
4. Illustrate the working of I/O operations and processing unit

Module-1	**9 Hours**

Introduction to Digital Design: Binary Logic, Axiomatic Definition of Boolean Algebra, Basic Theorems and Properties of Boolean Algebra, Boolean Functions, Digital Logic Gates

Gate Level Minimization: Introduction, The Map Method, Four-Variable Map, Don't-Care Conditions, NAND and NOR Implementation, Exclusive-OR Function

Hardware Description Language – Verilog Model of a simple circuit.

Textbook 1:Chapter-1(1.9), Chapter-2(2.3-2.5,2.8), Chapter-3(3.1- 3.3, 3.6-3.7, 3.9, 3.11)

Module-2	**9 Hours**

Combinational Logic: Introduction, Combinational Circuits, Design Procedure, Binary Adder- Subtractor, Decimal adder, Decoders, Encoders, Multiplexers.

HDL (Verilog) Models of Combinational Circuits – Adder, Multiplexer, Decoder.

Sequential Logic: Introduction, Sequential Circuits, Storage Elements: Latches, Flip-Flops- JK, Characteristic Tables

Textbook 1:Chapter-4(4.1-4.2, 4.4-4.6, 4.9-4.12), Chapter-5(5.1-5.4)

Module-3	**9 Hours**

Registers: Registers, 4 bit Register with parallel load, shift Register: 4 bit shift Register, Serial transfer, Serial Addition

Counters: Ripple Counters: Design of 4-Bit Binary ripple counter using D & T Flip-flops

Synchronous Counters: 4 bit Binary counter, 4 Bit Up/Down Binary counter, Design of Counter with Unused States

Textbook 1:Chapter-6(6.1- 6.5)

Module-4	**9 Hours**

Basic Structure of Computers: Functional Units, Basic Operational Concepts, Bus structure, Performance – Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement. **Machine Instructions and Programs:** Memory Location and Addresses, Memory Operations, Instruction and Instruction sequencing, Addressing Modes. **Text book 2: Chapter-1(1.2-1.4, 1.6), Chapter-2(2.2-2.5)**
Module-5 **9 Hours**
Input/output Organization: Accessing I/O Devices, Interrupts – Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Direct Memory Access: Bus Arbitration. Cache memory, Mapping functions- Direct, Associative, Set-associative **Basic Processing Unit: S**ome Fundamental Concepts- Register Transfers, Performing ALU operations, Fetching a word from Memory, Storing a word in memory. Execution of a Complete Instruction. **Text book 2:Chapter-4(4.1, 4.2.1, 4.2.2, 4.2.3, 4.4), Chapter-5(5.5, 5.5.1), Chapter-7(7.1-7.2)**

PRACTICAL COMPONENT OF IPCC

Sl. No.	EXPERIMENTS
	PART A
1	Realization of Logic gates AND, OR, NOT, NAND, NOR, XOR and XNOR using Verilog HDL
2	Given a 4-Variable logic expression, simplify it using appropriate technique and simulate the same using Basic gates in Verilog HDL.
3	Design Verilog HDL to implement simple circuits using Structural, Data flow and Behavioural model.
4	Design Verilog HDL to implement Half and Full Adder.
5	Design Verilog HDL to implement Half and Full Subtractor.
	PART B
6	Design Verilog HDL to implement 4 bit Binary Full adder, a 4 bit Binary Subtractor and simulate the same using basic gates.
7	Design Verilog HDL to implement Different types of Multiplexer - 2:1, 4:1 and 8:1.
8	Design Verilog HDL to implement Different types of De-Multiplexer- 1:2, 1:4 and 1:8
9	Design Verilog HDL for implementing various types of Flip-Flops such as JK and D
10	Design a 4 bit Ripple counter and implement with Verilog HDL
	Demo Experiment
1	Verilog implementation of AND, OR, and NOT logic gates using a MacCulloch-Pitts Perceptron model (single-layer Feedforward Neural Network).

COURSE OUTCOMES:

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

CO1	Apply the fundamental concepts of Binary logic, Boolean algebra, and logic gate minimization using Karnaugh Map (K-MAP) techniques.
CO2	Design and analyse Combinational and Sequential circuits.
CO3	Develop and evaluate the functionality of registers and counters.
CO4	Elaborate the fundamental structure and key performance parameters of a computer system.
CO5	Describe the concepts of Input/Output (I/O) organization and the process of instruction execution in a basic processing unit.

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.**
SEE (D)	**50**	**18**	**SEE is a theory exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall before 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. M. Morris Mano and Michael D. Ciletti, "Digital Design with an Introduction to Verilog Design", 5th Edition, Pearson Education, 2013.
2. Carl V. Hamacher, Zvonko G. Vranesic, and Safwat G. Zaky, "Computer Organization",5th Edition, Tata McGraw-Hill, 2002.
3. William Stallings, "Computer Organization and Architecture- designing for performance", 8th Edition, Prentice Hall, New Jersy, 2010.

Reference Books

1. M. Morris Mano and Charles R. Kime, "Logic and Computer Design Fundamentals", 4th Edition, Pearson Education, 2008.
2. David A. Patterson and John L. Hennessy, "Computer Organization and Design", 4th Edition, Morgan Kaufmann Publishers, 2009.
3. Anrew S. Tanenbaum, "Structured Computer Organization", 5th Edition, Pearson Education, 2006

Weblinks:

1. https://cse02-iiith.vlabs.ac.in/
2. https://ocw.mit.edu/courses/6-111-introductory-digital-systems-laboratory-fall-2006/
3. https://circuitverse.org/
4. https://www.geeksforgeeks.org/computer-organization-and-architecture-tutorials/
5. NPTEL : https://nptel.ac.in/courses/117105080
6. NPTEL: https://nptel.ac.in/courses/106102062

Self – Study topics:

- Finite State Machine (FSM) Design Using Verilog- Implementation of Mealy and Moore machines for real-time applications.
- RISC vs. CISC Architectures- Comparative study of instruction sets, execution efficiency, and real-world processor examples.
- Introduction to FPGA-Based Digital Design- Basics of Field Programmable Gate Arrays (FPGAs) and using HDL for hardware synthesis.
- Pipelining in CPU Design- Study of instruction pipeline stages, pipeline hazards, and performance improvement techniques.

Activity Based Learning

- Flipped Classroom
- Group Discussion (GD)

- Student Seminars
- Project Based Learning
- **Peer Teaching -** Make Groups and give crosswords related to the subject by using https://www.ohmydots.com/creator-crossword.html
- Hands-on Practical Session using IC Trainer Kit- To provide students with real-time exposure to implementing digital logic circuits using physical ICs and wiring on a trainer kit.

B.E. (Computer Science & Engineering) Outcome Based Education (OBE) and Choice Based Credit System (CBCS) SEMESTER – III			
Course Title:	**Operating Systems**		
Course Code:	**BCS303**	**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 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 to operating systems, System structures: What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations.

Operating System Services: User - Operating System interface; System calls; Types of system calls; System programs; Operating system design and implementation; Operating System structure.

Textbook 1: Chapter-1(1.1-1.5), Chapter-2(2.2-2.7)

Module-2 **9 Hours**

Process Management: Process concept; Process scheduling; Operations on processes; Inter process communication.

Multi-threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues.

Process Scheduling: Basic concepts; Scheduling Criteria; Scheduling Algorithms; Multiple-processor scheduling.

Textbook 1: Chapter-3(3.1-3.4), Chapter-4(4.1-4.4), Chapter-5(5.1 -5.3,5.5)

Module-3 **9 Hours**

Process Synchronization: Synchronization: The critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization.

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

Textbook 1: Chapter-6 (6.1-6.6), Chapter-7(7.1 -7.7)	
Module-4	**9 Hours**
Memory Management: Memory management strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. **Virtual Memory Management:** Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing. **Textbook 1: Chapter -8(8.1-8.6), Chapter 9(9.1-9.6)**	
Module-5	**9 Hours**
File System, Implementation of File System: File system: File concept; Access methods; Directory and Disk structure **Implementing File system:** File system structure; File system implementation; Directory implementation; Allocation methods; Free space management. **Secondary Storage Structure, Protection:** Mass storage structures; Disk structure; Disk Attachment, Disk scheduling; **Protection**: Goals of protection, Principles of protection, Domain of protection, Access matrix. **Textbook 1: Chapter-10(10.1-10.3) , Chapter-11(11.1-11.5), Chapter-12(12.1-12.4), Chapter-14(14.1-14.4)**	

PRACTICAL COMPONENT OF IPCC

EXPERIMENTS	
Sl. No	**PART A**
1.	Develop a C program to implement the Process system calls (fork (), exec (), wait(), create process, terminate process)
2.	Simulate the following CPU scheduling algorithms to find turnaround time and waiting time a) FCFS b) Round Robin
3.	Develop a C program to simulate producer-consumer problem using semaphores.
4.	Develop a C program which demonstrates interprocess communication between a reader process and a writer process. Use mkfifo, open, read, write and close APIs in your program.
5.	Develop a C program to simulate the Linked file allocation strategies
6.	Develop a C program to simulate SCAN disk scheduling algorithm.
PART B (Case Study Based Learning)	
1.	**CPU Scheduling – Priority, SJF, SRTF** A real-time OS for a medical monitoring device needs responsive CPU scheduling for patient alerts, logging, and UI display. **Questions:** • Given a set of processes with burst time and priorities, simulate execution under Priority Scheduling, SJF, and SRTF.

	• Which algorithm ensures minimum response time for emergency alerts? Can starvation occur in any algorithm? Propose a solution
2.	**Multithreading** You're developing a multithreaded application for online proctoring that must record video, monitor mouse movement, and detect tab switching concurrently. **Questions:** • Identify tasks that can run as separate threads. • How would you manage shared resources like logs or video buffers? • Design a synchronization scheme to avoid race conditions during thread execution.
3.	**Banker's Algorithm for Deadlock Avoidance** An automated research lab has robotic arms (resources) allocated to multiple experiment modules (processes). The system must avoid deadlocks when assigning robotic arms. **Questions:** • Given a snapshot of allocated and maximum resources, apply Banker's algorithm to determine if the system is in a safe state. • What should the system do if a new module requests more resources? • Modify the scenario to include resource preemption. How does this impact safety?
4.	**Dining Philosopher Problem** Your company is designing an automated restaurant system where robotic arms represent philosophers sharing limited cooking utensils (forks). **Questions:** • How will you model the utensil access to prevent deadlocks? • If one philosopher is replaced by a "VIP" who should never wait, how would you modify your synchronization logic? • Simulate one round of execution and analyze if any philosopher starves.
5.	**Contiguous Memory Allocation** A company is developing an embedded system for an industrial robot. The robot's operating system must load real-time control programs into memory using contiguous memory allocation for speed and predictability. **Questions:** • The control programs vary in size. How should the OS manage memory to minimize fragmentation? • Which allocation strategy (First Fit, Best Fit, Worst Fit) would be most efficient here and why? • After 10 load-unload cycles, analyze the state of memory. How can compaction help in this scenario?
6.	**Page Replacement Algorithms**

	An educational software company is optimizing a virtual memory system for an e-learning tablet. Students often multitask between notes, browser, and video lectures. **Questions:** • Simulate page accesses using FIFO, LRU, and Optimal algorithms. Compare page faults. • In real-time usage, which algorithm would you implement and why? • How will the choice of algorithm impact user experience when switching apps?
7.	**File Allocation** A cloud storage service stores small to medium files for academic institutions. Different strategies (contiguous, linked, indexed) impact retrieval time and fragmentation. **Questions:** • Given a set of file sizes and disk blocks, demonstrate how each method allocates space. • Which method performs best for random access vs. sequential access? • How will the system handle file growth in each case?
8.	**Directory Structure** Designing a multi-user OS for a campus intranet requires an efficient and secure directory structure. **Questions:** • Which directory structure (single-level, two-level, tree-structured) would best suit departmental file segregation? • How would you implement access permissions for faculty vs. students? • What are the limitations of a tree-structured directory if symbolic links are introduced?
9.	**Disk Scheduling Algorithms** A university server handles high-volume read/write requests for students accessing online lab submissions. **Questions:** • Using a queue of 10 disk I/O requests (randomly ordered), simulate FCFS, SSTF, and SCAN algorithms. • Which algorithm minimizes seek time and why? • What trade-offs would you consider for fairness vs. efficiency?
10.	Build a simplified version of the ls command using only system calls (no standard library functions like opendir()).

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.

Textbooks

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, "Operating System Concepts", 8th 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.

Weblinks

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=dOiA2nNJpc0

Self-Study 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, Unix, Android, MacOS, iOS, RTOS, Chrome OS, Solaris, Haiku OS, TinyOS

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.
SEE (D)	50	18	SEE is a theory exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall before 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.

Alternate Assessment Methods

1. **Certifications (Skill-Based) -**industry-recognized certifications (e.g. CISCO, NPTEL, Infosys Springboard).
2. **Assignments / Problem Solving Sheets**
3. **Quiz**
4. **Mini Projects**
5. **Group Activities / Peer Evaluation**
6. **Presentations / Tech Talks**

B.E. (Computer Science and Engineering) Outcome Based Education(OBE) and Choice Based Credit System(CBCS) SEMESTER – III			
Course Title:	**Data Structures and Applications**		
Course Code:	**BCS304**	**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 fundamental concepts, classifications, and operations of data structures.
2. Implement arrays, pointers, and strings for efficient data representation and manipulation.
3. Apply linear data structures such as stacks, queues, and linked lists to solve problems.
4. Construct and traverse non-linear data structures including trees and graphs.
5. Utilize advanced data structures like hashing and balanced trees to design optimized solutions.

Module-1	**9 Hours**
Introduction To Data Structures: Data Structures, Classifications (Primitive & Non-Primitive), Data structure Operations, Review of pointers and dynamic Memory Allocation **Arrays and Structures:** Arrays, Sorting using arrays: Insertion sort, Radix sort, Dynamic Allocated Arrays, Structures and Unions, Polynomials, Sparse Matrices, representation of Multidimensional Arrays, Strings **Textbook 1: Chapter-2 (2.4-2.5)** **Textbook 2: Chapter-2 (2.1-2.3), Chapter-3 (3.1-3.13), Chapter-4 (4.1-4.4), Chapter-5 (5.1-5.8), Chapter-14(14.8-14.12)**	
Module-2	**9 Hours**
Stacks: Stacks, Stacks Using Dynamic Arrays, Evaluation and conversion of Expressions, Recursion **Queues:** Queues, Circular Queues using Dynamic Arrays, Priority Queue, Multiple Stacks and queues. **Textbook 1:Chapter-3 (3.1- 3.7)**	
Module-3	**9 Hours**
Linked Lists: Singly Linked, Lists and Chains, Representing Chains in C, Linked Stacks and Queues, Polynomials, Additional List Operations, Sparse Matrices, Doubly Linked List. **Textbook 1:Chapter-4 (4.1- 4.5, 4.7-4.8)**	
Module-4	**9 Hours**
Trees: Introduction, Binary Trees, Binary Tree Traversals, Threaded Binary Trees., Binary Search trees, Selection Trees, Forests. **Textbook 1:Chapter-5 (5.1-5.3, 5.5, 5.7-5.9)**	
Module-5	**9 Hours**
Multiway Search Trees: B-Trees, B+-Trees	

Graphs: The Graph Abstract Data Types, Elementary Graph Operations. **Hashing:** Introduction, Static Hashing, Dynamic Hashing **Textbook 1: Chapter-6 (6.1-6.2), Chapter -8 (8.1 - 8.3), Chapter -9 (9.1-9.2), Chapter -11(11.2,11.3)**

COURSE OUTCOMES:

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

CO1	Implement fundamental data structures and perform operations using arrays, pointers, and strings.
CO2	Apply stacks and queues to evaluate expressions and solve linear data structure problems.
CO3	Develop memory-efficient programs using singly and doubly linked lists.
CO4	Construct and traverse tree-based structures to model and analyse hierarchical relationships.
CO5	Design solutions using graphs, balanced trees, and hashing techniques to address real-world computational problems.

Textbooks

1. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, "Fundamentals of Data Structures in C", 2nd Edition, Universities Press, 2014.
2. Reema Thareja, "Data Structures using C", 2nd Edition, Oxford press, 2014.

Reference Books

1. Seymour Lipschutz, "Data Structures Schaum's Outlines", Revised 1st Edition, McGraw Hill, 2014
2. Gilberg & Forouzan, "Data Structures: A Pseudo-code approach with C", 2nd Edition, Cengage Learning,2014
3. Jean-Paul Tremblay & Paul G. Sorenson, "An Introduction to Data Structures with Applications", 2nd Edition, McGraw Hill, 2013
4. A M Tanenbaum, "Data Structures using C", PHI, 1989

Web links and Video Lectures (e-resources):

1. https://youtu.be/S47aSEqm_0I?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-traversal/dft-practice.html
8. https://onlinecourses-archive.nptel.ac.in/noc18_cs25/preview

Self-Study Topics:

Recursion, AVL Trees, Red Black Trees.

Activity Based Learning: The following is shown as examples only but not limited to:

1. LeetCode - The World's Leading Online Programming Learning Platform
2. HackerRank - Online Coding Tests and Technical Interviews

ASSESSMENT DETAILS (Both CIE and SEE)

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 before 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.			

Continuous and Comprehensive Evaluation (B)

Component	Weightage	Evaluation Method
HackerRank / LeetCode Problems	10 Marks	- Solve 2–3 DS problems per module. - Submit screenshots + short explanation. - Marks based on test cases passed + clarity.
MOOC Certification	10 Marks	- Complete an approved MOOC (NPTEL / Coursera / edX). - Submit certificate + short reflective report. - Marks based on completion
Total	**20 Marks**	Alternate Assessment (CIE)

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

Course Title:	**Data Structures Laboratory**		
Course Code:	**BCSL305**	**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 and implement dynamic memory management techniques in C.
2. Apply linear data structures (stacks, queues, linked lists) to solve computational problems.
3. Analyse and utilize non-linear data structures such as trees and graphs for applications.
4. Design practical solutions to real-world problems using suitable data structures.

Program 1 – Weekly Planner

A student needs a simple Weekly Planner System to keep track of their weekly schedule. For each day, the planner should store: name of the day (e.g., Monday, Tuesday…), date of the day (numeric format) and the activity planned for that day (e.g., class, event, meeting).

Develop a C program to implement this weekly planner using the following functions.

a. create() to initialize the weekly planner.
b. read() to input the details (day name, date, activity) from the user.
c. display() to generate and print the Weekly Activity Report on screen.

Program 2 –Pattern Matching

A publishing company is developing a simple tool to edit documents by replacing certain words or phrases. Design a C program that performs find-and-replace operations on text without using built-in string functions.

a. Read a main string (STR), a pattern string (PAT), and a replacement string (REP) from the user.
b. Search the main string for all occurrences of PAT.
c. Replace every occurrence of PAT in STR with REP.
d. If PAT does not exist in STR, display an appropriate message.

Program 3- Parcel Storage System

A courier office needs a simple Parcel Storage System to manage parcels efficiently. Each parcel is assigned an integer ID for identification. Design a C program that supports the following operations:

a. push() – Add a parcel ID onto the stack when a new parcel arrives.
b. pop() – Remove the last parcel ID from the stack when a parcel is taken out.
c. display() – Show all parcel IDs currently stored in the stack.
d. Display suitable messages when the stack is full (overflow) or empty (underflow).

Program 4 - Calculator Application

In modern calculators, mathematical expressions are usually entered in infix notation, where operators are written between operands (e.g., 3 + 4 * 2). However, calculators and computer systems evaluate these expressions more efficiently in postfix notation, where operators follow operands (e.g., 3 4 2 * +).

To simulate this process, develop a C program that converts any valid infix expression into its equivalent postfix expression. The program should support alphanumeric operands and the following operators: -, +, *, /, %, ^.

Program 5 - Call Center Help Desk

A call center assigns a ticket number to every incoming call. Since only a limited number of calls can be handled at a time, the calls are placed in a queue. When a representative becomes free, the next call in the queue is served.

Design and implement a C program to simulate this system using a circular queue. The program should support the following operations:

a. Add a new call to the queue (enqueue)
b. Serve the next call (dequeue)
c. Display all waiting calls

The program must also handle queue overflow and underflow conditions properly.

Program 6 - Student Information Management System

A university wants to maintain a dynamic record of its students for academic and administrative purposes. Each student record contains the following details: **USN, Name, Programme, Semester, and Phone Number**. Since the number of students may vary frequently due to admissions and withdrawals, a **singly linked list (SLL)** is chosen for efficient memory usage and flexible data management.

To simulate this system, develop a **menu-driven C program** that performs the following operations on the student database:

a. **Create** a Singly Linked List of *N* student records using **front insertion**.
b. **Display** all student records and **count** the number of nodes in the list.
c. **Insert** or **delete** a student record at the **end** of the list.
d. **Insert** or **delete** a student record at the **front** of the list (demonstrating **stack operations**).

Program 7 - Employee Record Management System

A company maintains details of its employees for administrative and payroll purposes. Each employee record contains the following fields: SSN, Name, Department, Designation, Salary, and Phone Number. Since employees may frequently join, leave, or change positions, a Doubly Linked List (DLL) provides a flexible way to manage these records efficiently in both directions.

To simulate this process, design and implement a menu-driven C program that performs the following operations on the employee database:

a. Create a Doubly Linked List of *N* employee records using end insertion.
b. Display all employee records and count the total number of nodes.
c. Insert or delete an employee record at the end of the list.
d. Insert or delete an employee record at the front of the list.
e. Demonstrate how the DLL can be used as a Double Ended Queue (Deque).

Program 8 – Library Book Management System

Design and implement a menu-driven C program to manage a Library Book Database using a Binary Search Tree (BST). Each record should store the following details: ISBN Number and Book Title.

The program should support the following operations:

a. Insert a new book into the BST
b. Traverse the BST using Inorder, Preorder, and Postorder
c. Search for a book by its ISBN number and print the title.

Program 9 - City Connectivity using Graph Traversal

In a city transportation system, each city can be represented as a node and the direct roads between them as edges of a graph. To analyse connectivity, it is important to know which cities can be reached from a given starting city.

Develop a C program to:

a. Represent N cities using an Adjacency Matrix
b. Display all cities reachable from a specified city using DFS or BFS traversal

Program 10 - Collision Handling in Vehicle Registration System

In a vehicle registration system, each vehicle is assigned a unique 4-digit registration number. To support quick storage and retrieval, these records are organized in a hash table with m storage locations.

Develop a C program to:

1. Store vehicle registration numbers in the hash table using linear probing and quadratic probing for collision handling.
2. Display the final state of the hash table after all insertions.

Mini Project

a. Students must form teams of 2 to 3 members.
b. Each team must identify and finalize a real-world problem that can be solved using appropriate data structures.
c. Teams must analyze the problem and design the solution using suitable data structures such as arrays, stacks, queues, linked lists, trees, or graphs.
d. The project must be implemented entirely in C language.
e. Modular programming practices, file handling, and inclusion of at least one innovative or optimized component are mandatory.
f. The complete project including source code, README file with instructions, and sample output screenshots must be uploaded to a public GitHub repository.
g. A mini project report (soft bound) must be prepared and submitted as per department guidelines.
h. Teams will present their project and undergo a viva voce as part of the final evaluation

- Smart India Hackathon : Smart India Hackathon (sih.gov.in)
- ISRO Hackathon: Launching of Bharatiya Antariksh Hackathon-2025 (isro.gov.in)
- DRDO Hackathon: Dare to Dream 5.0 Innovation Contest | Defence Research and Development Organisation - DRDO, Ministry of Defence, Government of India

COURSE OUTCOMES

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

CO1	Implement and demonstrate operations on fundamental data structures
CO2	Develop programs to solve problems using linear and non-linear data structures.
CO3	Apply dynamic allocation and optimizing resource usage.
CO4	Design and implement practical solutions for real-world problems using appropriate data structures.

ASSESSMENT DETAILS (Both CIE and SEE)

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE Practical (A)	20	-	Marks awarded for Weekly Conduction of Experiments and Submission of Laboratory records, scaled to 20 marks.
CIE Practical Test + Mini Project (B)	20+10	-	Average of marks of Two tests, each conducted for 100 marks covering all experiments, scaled to 20 + Mini Project (10 Marks)

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.			

B.E. (Computer Science and Engineering) Outcome Based Education (OBE) and Choice Based Credit System (CBCS) SEMESTER – III			
Course Title:	**Object Oriented Programming using Java**		
Course Code:	**BCS306A**	**CIE MARKS**	**50**
(L: T: P) + SL	**(2:0:2) + (30 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. Identify and recall core OOP concepts and Java language features.
2. Configure the Java development environment and execute simple Java programs.
3. Develop multi-threaded Java applications using synchronization methods.
4. Design interactive Java applications using event-driven programming principles

Module-1	**6 Hours**
Introducing Classes: Class Fundamentals, Declaring Objects, Assigning Object Reference Variables, Introducing Methods, Constructors, This Keyword, Garbage Collection. **Methods and Classes**: Overloading Methods, Objects as Parameters, Argument Passing, Returning Objects, Recursion, Access Control, understanding static, introducing final, introducing Nested and Inner Classes. **Textbook 1: Chapter-6, Chapter-7**	
Module-2	**6 Hours**
Inheritance: Inheritance Basics, using super, creating a Multilevel Hierarchy, When Constructors Are Executed, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using final with Inheritance, Local Variable, Type Inference and Inheritance, The Object Class. **Textbook 1: Chapter-8**	
Module-3	**6 Hours**
Exceptions: Exception-Handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch, Multiple catch Clauses, Nested try Statements, throw, throws, finally, Java's Built-in Exceptions, Creating Your Own Exception Subclasses. **Textbook 1: Chapter-10**	
Module-4	**6 Hours**
Multithreaded Programming: The Java Thread Model, The Main Thread, creating a Thread, Creating Multiple Threads, Using isAlive() and join(), Thread Priorities, Synchronization, Interthread Communication, Suspending, Resuming, and Stopping Threads. **Textbook 1: Chapter-11**	
Module-5	**6 Hours**
String Handling: The String Constructors, String Length, Special String Operations, Character Extraction, String Comparison, Searching Strings, modifying a String, Changing the Case of Characters Within a String, Joining Strings. **Textbook 1: Chapter-18**	

PRACTICAL COMPONENT

Sl. No.	Experiments
1	Develop a JAVA program to add TWO matrices of suitable order N (The value of N should be read from command line arguments).
2	Develop a stack class to hold a maximum of 10 integers with suitable methods. Develop a JAVA main method to illustrate Stack operations.
3	A class called Employee, which models an employee with an ID, name and salary, is designed as shown in the following class diagram. The method raiseSalary (percent) increases the salary by the given percentage. Develop the Employee class and suitable main method for demonstration
4	Develop a JAVA program to create a class named shape. Create three sub classes namely: circle, triangle and square, each class has two member functions named draw () and erase (). Demonstrate polymorphism concepts by developing suitable methods, defining member data and main program.
5	Develop a JAVA program to create an abstract class Shape with abstract methods calculateArea() and calculatePerimeter(). Create subclasses Circle and Triangle that extend the Shape class and implement the respective methods to calculate the area and perimeter of each shape.
6	Develop a JAVA program to create an interface Resizable with methods resizeWidth(int width) and resizeHeight(int height) that allow an object to be resized. Create a class Rectangle that implements the Resizable interface and implements the resize methods
7	Develop a JAVA program to create an outer class with a function display. Create another class inside the outer class named inner with a function called display and call the two functions in the main class.
8	Develop a JAVA program to raise a custom exception (user defined exception) for DivisionByZero using try, catch, throw and finally.
9	Write a program to illustrate creation of threads using runnable class. (start method start each of the newly created thread. Inside the run method there is sleep() for suspend the thread for 500 milliseconds).
10	Implement a JAVA program to illustrate the use of different types of character extraction, string comparison, string search and string modification methods.
11	Develop a Java application that checks whether a given string is a palindrome, designed using the SOLID principles of object-oriented programming.
12	Develop a Java application that compresses characters in a given string by counting consecutive repeated characters, designed according to the SOLID principles of object-oriented programming. Input: AAABBBCCCC Output: A3B3C4

COURSE OUTCOMES

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

CO1	Interpret Java classes with appropriate data members and methods to address specified scenarios.
CO2	Apply the concept of inheritance to model and solve real-world programming problems.
CO3	Demonstrate the effective use of multi-threading and exception handling to manage concurrency and runtime issues in Java programs.
CO4	Demonstrate the use of string handling methods to manipulate and process textual data in Java.
CO5	Develop Java programs integrating object-oriented concepts to address computational problems effectively.

ASSESSMENT AND EVALUATION REGULATIONS OF ESC

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.**
SEE (D)	**50**	**18**	**SEE is a theory exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall before 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 ESC shall be for CIE only. However, in SEE, the questions from the laboratory component shall be included in their respective modules only.			

Textbooks

1. Herbert Schildt, "The Complete Reference", 12th Edition, McGraw-Hill.

Reference Books

1. E Balagurusamy,"Programming with Java", 6th Edition, McGraw Hill.
2. Bruce Eckel, "Thinking in Java", 4th Edition, Prentice Hall.

Web links and Videos:

1. https://www.youtube.com/watch?v=5gWpVJc0UG4
2. https://www.youtube.com/watch?v=-HafzawNlUo
3. https://www.youtube.com/watch?v=K3g4srbkUNM
4. https://www.youtube.com/watch?v=iUzvdDuAeSQ

Self-Study Topics:

SOLID Principles-GFG

Activity Based Learning:

1. Leet code
2. Practical or programming assignment
3. Seminar/Project

B.E. (Computer Science and Engineering) Outcome Based Education (OBE) and Choice Based Credit System (CBCS) SEMESTER – III			
Course Title:	**Introduction to Data Science**		
Course Code:	**BCS306B**	**CIE MARKS**	**50**
L: T: P	**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. Acquire fundamental knowledge on Data Science with querying and analytics required for the field of Data Science.
2. Understand the process of handling heterogeneous data, pre-process and visualize them for better understanding.
3. Analyse basic tools of EDA, Data Science process with case studies and Different algorithms.
4. Gain the fundamental knowledge on Data Science techniques.
5. Explore Map reduce and data visualization.

Module-1 **9 Hours**

Introduction to 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

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

Module-2 **9 Hours**

Statistical Inference: Statistical Inference, Populations and samples, Populations and samples of Big Data.

Exploratory Data Analysis, the Data Science Process: Basic tools (plots, graphs and summary statistics) of EDA, Philosophy of EDA, Case Study: Real Direct (online real estate firm). Basic Machine Learning Algorithms: Linear Regression, k-Nearest Neighbours (kNN), k-means.

Module-3 **9 Hours**

Spam Filter using Linear Regression, Spam Filter using K-NN, Naïve Bayes Algorithm, Spam Filter using Naïve Bayes, Laplace Smoothing, Comparing Naïve Bayes to K-NN, Scraping the Web.

Logistic Regression: Introduction to Logistic Regression, Thought experiments, Classifiers, M6D case study.

Module-4 **9 Hours**

Extracting meaning from data: Background- Data Science competitions, crowd sourcing, The Kaggle model, Thought Experiment: What are the ethical implications of a Robo-grader? Feature Selection algorithms. Example-User Retention, Filters; Wrappers; Decision Trees; Random Forests. User Retention, Google's Hybrid approach to social research, Thought Experiment: What is the best way to decrease concern and increase understanding and control? **Recommendation Systems: Building a User-Facing Data Product at Scale**, A Real-World Recommendation Engine, Nearest Neighbour Algorithm review, some problems with Nearest Neighbours, Beyond Nearest Neighbour- Machine Learning Classification, the dimensionality problem, Singular Value Decomposition, Important properties of SVD, Principal Component Analysis, Alternating Least Squares.
Module-5 **9 Hours**
Data engineering: Data Engineering, Map reduce, Word Frequency Problem, Map Reduce Solution, Other Examples of Map Reduce, Pregel **Data Visualization and Fraud Detection:** Data visualization History, what is Data Science, Redux? A sample visualization projects, Data Science and risk, Data visualization at square.

COURSE OUTCOMES:

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

CO1	Recall and explain the fundamental concepts and real-world applications of Data Science, and make inferences from key concepts.
CO2	Demonstrate understanding of the process of Exploratory Data Analysis (EDA) and the Data Science workflow.
CO3	Explain and apply basic Machine Learning techniques for problem solving.
CO4	Compare and analyse technical methods such as feature selection and
CO5	Explore and implement MapReduce concepts and apply data visualization techniques for insights.

ASSESSMENT DETAILS (Both CIE and SEE)

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, Anuradha D. Thakare, "Fundamentals of Data Science", CRC Press, 1st Edition, 2021.
2. Cathy O'Neil, Rachel Schutt, "Doing Data Science: Straight Talk from the Frontline", O'Reilly Media, 1st Edition, 2013.
3. Joel Grus, "Data Science from Scratch: First Principles with Python", O'Reilly Media, 2nd Edition, 2019

Reference Books:

1. Kevin P. Murphy, "Machine Learning: A Probabilistic Perspective", MIT Press, 1st Edition, 2012.
2. Peter Bruce, Andrew Bruce, "Practical Statistics for Data Scientists", O'Reilly Media, 1st Edition, 2017.
3. Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, "An Introduction to Statistical Learning with Applications in R", Springer, 2nd Edition, 2021.

B.E. (Computer Science and Engineering) Outcome Based Education (OBE) and Choice Based Credit System (CBCS) SEMESTER – III			
Course Title:	**Data Analytics with R**		
Course Code:	**BCD306C**	**CIE MARKS**	**50**
(L :T : P) + SL	**2:0:2 + (30 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 Gain the knowledge of R Programming Concepts
2. To Explain the concepts of Data Visualization
3. To Explain the concept of Statistics in R.
4. To Work with R charts and Graphs

Module-1	**6 Hours**

Introduction to R programming: Introducing R, Initiating R, Packages in R, Environments and Functions, Flow Controls, Loops, Basic Data Types in R, Vectors.

Textbook 1 : Chapter-1(1.1-1.7),Chapter -2(2.1-2.2)

Module-2	**6 Hours**

Data Structures in R : Matrices and Arrays, Lists, Data Frames, Factors, Strings, Dates and Times ,Time Zones, Calculations with Dates and Times.

Data Preparation: Datasets, Importing and Exporting files, Accessing Databases, Data Cleaning and Transformation.

Textbook 1: Chapter-2(2.3-2.8),(2.8.1-2.8.4),Chapter-3(3.1-3.4)

Module-3	**6 Hours**

Charts, Plots, and Visual Tools for Understanding Data Patterns : Graphics using R , Exploratory Data Analysis, Main Graphical Packages, Pie Charts, Scatter Plots, Line Plots, Histograms, Box Plots, Bar Plots, Other Graphical packages.

Textbook 1: Chapter-4(4.1-4.9)

Module-4	**6 Hours**

Statistical Analysis Using R: Statistical Measures, Normal distribution, Binomial distribution, Correlation Analysis, Regression Analysis-Linear Regression Analysis of Variance.

Textbook 1: Chapter-5(5.1, 5.3, 5.4-5.6.1, 5.7)

Module-5	**6 Hours**

Data Mining and Pattern Discovery Using R: Packages for Data Mining, Clustering using R,K means Clustering, Classification-Packages and functions for classification, Decision Tree, Outlier detection Using R.

Textbook 1: Chapter-6(6.1,6.2.2,6.3,6.3.2,6.5,6.5.2-6.5.4)

PRACTICAL COMPONENT

Program 1. Introduction to R Programming: Data Types, Operations, and Basic Data Structures

Demonstrate the steps for installation of R and R Studio. Perform the following:

a. Assign different type of values to variables and display the type of variable. Assign different types such as Double, Integer, Logical, Complex and Character and understand the difference between each data type.
b. Demonstrate Arithmetic and Logical Operations with simple examples.
c. Demonstrate generation of sequences and creation of vectors.
d. Demonstrate Creation of Matrices
e. Demonstrate the Creation of Matrices from Vectors using Binding Function.
f. Demonstrate element extraction from vectors, matrices and arrays.

Program 2. Financial Performance Analysis in R using Vectors: Monthly Profit Evaluation and Reporting

Assess the Financial Statement of an Organization being supplied with 2 vectors of data: Monthly Revenue and Monthly Expenses for the Financial Year. You can create your own sample data vector for this experiment) Calculate the following financial metrics:

a. Profit for each month.
b. Profit after tax for each month (Tax Rate is 30%).
c. Profit margin for each month equals to profit after tax divided by revenue.
d. Good Months – where the profit after tax was greater than the mean for the year.
e. Bad Months – where the profit after tax was less than the mean for the year.
f. The best month – where the profit after tax was max for the year.
g. The worst month – where the profit after tax was min for the year.

Note: a. All Results need to be presented as vectors, b. Results for Dollar values need to be calculated with $0.01 precision, but need to be presented in Units of $1000 (i.e 1k) with no decimal points ,c. Results for the profit margin ratio need to be presented in units of % with no decimal point,d. It is okay for tax to be negative for any given month (deferred tax asset) ,e. Generate CSV file for the data.

Program 3. Matrix Operations in R: Implementation of Basic Arithmetic and Transpose on 3x3 Matrices

Develop a program to create two 3 X 3 matrices A and B and perform the following operations

a. Transpose of the matrix
b. addition
c. subtraction
d. multiplication
e. e) Division

Program 4. Prime Number Generation in R Using Sieve of Eratosthenes with Functions

Develop an R Program using functions to find all the prime numbers up to a specified number by the method of Sieve of Eratosthenes.

Program 5. Leap Year Checker in R Using Conditional Statements

Implement R script to check the given year is leap year or not.

Program 6. String Manipulation in R: Exploring Text Processing Functions

Implement different String Manipulation functions in R.

Program 7. Correlation Analysis and Visualization Using the 'mammals' Dataset in R

The built-in data set mammals contain data on body weight versus brain weight.
Develop R commands to:

a. Find the Pearson and Spearman correlation coefficients. Are they similar?
b. Plot the data using the plot command.
c. Plot the logarithm (log) of each variable and see if that makes a difference.

Program 8. Data Frame Creation, Sub-setting, and Merging in R for Inventory Analysis

Develop R program to create a Data Frame with following details and do the following operations.

a. Subset the Data frame and display the details of only those items whose price is greater than or equal to 350.
b. Subset the Data frame and display only the items where the category is either “Office Supplies” or “Desktop Supplies”
c. Create another Data Frame called “item-details” with three different fields itemCode, ItemQtyonHand and ItemReorderLvl and merge the two frames

itemCode	itemCategory	itemPrice
1001	Electronics	700
1002	Desktop Supplies	300
1003	Office Supplies	350
1004	USB	400
1005	CD Drive	800

Program 9. Customized Histogram Visualization in R Using the 'airquality' Dataset

Let us use the built-in dataset air quality which has Daily air quality measurements in New York, May to September 1973. Develop R program to generate histogram by using appropriate arguments for the following statements.

a. Assigning names, using the air quality data set.
b. Change colors of the Histogram
c. Remove Axis and Add labels to Histogram
d. Change Axis limits of a Histogram
e. Add Density curve to the histogram

Program 10. Employee Data Analysis in R: Importing, Filtering, and Exporting CSV Files

Design a data frame in R for storing about 20 employee details. Create a CSV file named "input.csv" that defines all the required information about the employee such as id, name, salary, start_date, dept. Import into R and do the following analysis.

a. Find the total number rows & columns
b. Find the maximum salary
c. Retrieve the details of the employee with maximum salary
d. Retrieve all the employees working in the IT Department.
e. Retrieve the employees in the IT Department whose salary is greater than 20000 and write these details into another file "output.csv".

Program 11. Descriptive Statistics and Correlation Analysis of the 'mtcars' Dataset in R

Using the built in dataset mtcars which is a popular dataset consisting of the design and fuel consumption patterns of 32 different automobiles. The data was extracted from the 1974 Motor Trend US magazine, and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles (1973-74 models). Format A data frame with 32 observations on 11 variables : [1] mpg Miles/(US) gallon, [2] cyl Number of cylinders [3] disp Displacement (cu.in.), [4] hp Gross horsepower [5] drat Rear axle ratio,[6] wt Weight (lb/1000) [7] qsec 1/4 mile time, [8] vs V/S, [9] am Transmission (0 = automatic, 1 = manual), [10] gear Number of forward gears, [11] carb Number of carburetors Develop R program, to solve the following:

a. What is the total number of observations and variables in the dataset?
b. Find the car with the largest hp and the least hp using suitable functions
c. Plot histogram / density for each variable and determine whether continuous variables are normally distributed or not. If not, what is their skewness?
d. What is the average difference of gross horse power(hp) between automobiles with 3 and 4 number of cylinders(cyl)? Also determine the difference in their standard deviations.
e. Which pair of variables has the highest Pearson correlation?

Program 12. Simple Linear Regression in R: Modeling Salary Progression with Experience

Demonstrate the progression of salary with years of experience using a suitable data set (You can create your own dataset). Plot the graph visualizing the best fit line on the plot of the given data points. Plot a curve of Actual Values vs. Predicted values to show their correlation and performance of the model. Interpret the meaning of the slope and y-intercept of the line with respect to the given data. Implement using lm function. Save the graphs and coefficients in files. Attach the predicted values of salaries as a new column to the original data set and save the data as a new CSV file.

COURSE OUTCOMES:

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

CO1	Understand the Structures of R Programming.
CO2	Illustrate the Basics of Data Preparation Using Real-World Examples.
CO3	Apply Graphical Packages in R for Data Visualization.
CO4	Apply Various Statistical Analysis Methods for Data Analytics.
CO5	Integrate R with Machine Learning Libraries for Predictive Analytics.

ASSESSMENT DETAILS (Both CIE and SEE)

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.(Mini projects and presentation).
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 out of allotted 50 marks to appear for SEE, with minimum 40 % scored in every component of CIE. 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. G. Sudhamathy and C. Jothi Venkateswaran, "R Programming: An Approach to Data Analytics", MJP Publishers, 2019

Reference Books

1. Cotton, R,"Learning R: A Step-by-Step Function Guide to Data Analysis". 1st Edition, O'Reilly Media Inc,2013
2. W. N. Venables, D.M. Smith and the R Development Core Team, "An Introduction to R, Notes on R: A Programming Environment for Data Analysis and Graphics", Ver 3.0.1 (2013-05-16)

Web links and Video Lectures (e-Resources):

1. URL: **https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf**
2. **http://www.tutorialspoint.com/r/r_tutorial.pdf 3**.
3. https://users.phhp.ufl.edu/rlp176/Courses/PHC6089/R_notes/intro.html
4. https://cran.r-project.org/web/packages/explore/vignettes/explore_mtcars.html 5.
5. https://www.w3schools.com/r/r_stat_data_set.asp 6. **https://rpubs.com/BillB/217355**

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- Project based Assignment

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

Course Title:	**Haskell Programming**		
Course Code:	**BCS306D**	**CIE MARKS**	**50**
L : T : P	**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 functional programming and how it differs from imperative paradigms.
2. Learn Haskell syntax, semantics, and its core concepts like immutability and recursion.
3. Apply higher-order functions and polymorphism in problem-solving.
4. Develop interactive Haskell programs using I/O and modules.

Module-1	**9 Hours**
Introduction to Functional Programming and Haskell Basics: Introduction to Functional Programming, Functions and Evaluation, Expressions, Types, and Definitions, Introduction to GHCi and Haskell Environment, Basic Data Types (Bool, Int, Float, Char, String), Modules and Libraries.	
Module-2	**9 Hours**
Function Design and Recursion: Designing Programs in Haskell, Local Definitions and Guards, Recursive Function Design and Strategies, Enumerated Types and Simple Algebraic Data Types, Basic Program Testing**.**	
Module-3	**9 Hours**
Working with Lists and Tuples: Tuple Types and Pattern Matching, Introduction to Lists and List Comprehensions, Programming with Lists: Map, Filter, Fold, Standard List Functions in Prelude, Polymorphism and Type Inference.	
Module-4	**9 Hours**
Input/Output and Interactive Programs: I/O Concepts in Haskell, The IO Type and do Notation, Simple Console Interactions, Case Study: Rock-Paper-Scissors, Error Handling and Recursion in I/O.	
Module-5	**9 Hours**
Higher-Order Functions and Functional Design: Functions as Arguments and Results, Lambda Abstractions and Currying, Partial Application, Building Higher-Order Programs, Case Studies (e.g., Pictures, Supermarket Billing**).**	

COURSE OUTCOMES:

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

CO1	Explain functional programming concepts and Haskell basics.
CO2	Design and implement recursive functions and algebraic data types.
CO3	Apply pattern matching, list comprehensions, and list functions for data manipulation.
CO4	Develop interactive console applications using I/O and error handling.
CO5	Analyze and build higher-order functions and functional designs for real-world scenarios.

ASSESSMENT DETAILS (Both CIE and SEE)

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. Simon Thompson, "Haskell: The Craft of Functional Programming", Addison-Wesley, 3rd Edition, 2011.
2. Graham Hutton, "Programming in Haskell", Cambridge University Press, 2nd Edition, 2016.
3. Paul Hudak, John Peterson, Joseph Fasel, "A Gentle Introduction to Haskell", 2000.

Reference Books:

1. Jeremy Gibbons, Richard Bird, "Algorithm Design with Haskell", Cambridge University Press, 1st Edition, July 2020.
2. Alejandro Serrano Mena, "Practical Haskell: A Real-World Guide to Programming", Apress, 2nd Edition, April 2019.

Miran Lipovača, "Learn You a Haskell for Great Good!: A Beginner's Guide", No Starch Press, 1st Edition, 2011.

B.E. (Common to All Branches)
Outcome Based Education(OBE) and Choice Based Credit System(CBCS)
SEMESTER – III

Course Title:	**Social Connect and Responsibility**		
Course Code:	**BSCK307**	**CIE MARKS**	**100**
L :T : P	**0:0:2**	**SEE MARKS**	**-**
Credits:	**1**	**EXAM HOURS**	**-**

COURSE LEARNING OBJECTIVES:

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

1. Provide a formal platform for students to communicate and connect to the surrounding.
2. Create a responsible connection with the society.
3. Understand the community in general in which they work.
4. Identify the needs and problems of the community and involve them in problem – solving.
5. Develop among themselves a sense of social & civic responsibility & utilize their knowledge in finding practical solutions to individual and community problems.
6. Develop competence required for group-living and sharing of responsibilities & gain skills in mobilizing community participation to acquire leadership qualities and democratic attitudes.

General Instructions - Pedagogy:

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. In addition to the traditional lecture method, different types of innovative teaching methods may be adopted so that the activities will develop students' theoretical and applied social and cultural skills.
2. State the need for activities and its present relevance in the society and provide real-life examples.
3. Support and guide the students for self-planned activities.
4. You will also be responsible for assigning homework, grading assignments and quizzes, and documenting students' progress in real activities in the field.
5. Encourage the students for group work to improve their creative and analytical skills.

Module-1 **3 Hours**

Plantation and adoption of a tree: Plantation of a tree that will be adopted for four years by a group of BE students. (ONE STUDENT- ONE TREE).

Students can prepare either a Report or a Poster describing the plant's origin, its usage in daily life, its appearance in folklore and literature.

Module-2 **3 Hours**

Heritage walk and crafts corner

Heritage walk- Heritage tour, knowing the history and culture of the city, connecting to people around through their history.

crafts corner- Knowing the city and its craftsman, photo blog and documentary on evolution and practice of various craft forms.
Module-3 **3 Hours**
Organic farming and waste management **Organic Farming:** History of organic farming, objective of organic farming, types of organic farming, Usefulness of organic farming. **Waste Management:** Study of wet waste management in college, Hostel and neighboring villages, and implementation in the campus.
Module-4 **3 Hours**
Water conservation Knowing the present practices in the surrounding villages and implementation in the campus, documentary or photo blog presenting the current practices.
Module-5 **3 Hours**
Food walk City's culinary practices, food lore, and indigenous materials of the region used in cooking.

Course outcomes

At the end of the course, the student will be able to:
CO1: Communicate and connect to the surrounding.
CO2: Create a responsible connection with the society.
CO3: Involve in the community in general in which they work.
CO4: Notice the needs and problems of the community and involve them in problem –solving.
CO5: Develop among themselves a sense of social & civic responsibility & utilize their knowledge in finding practical solutions to individual and community problems.
CO6: Develop competence required for group-living and sharing of responsibilities & gain skills in mobilizing community participation to acquire leadership qualities and democratic attitudes.

Assessment Details for CIE (both CIE and SEE)

Weightage	**CIE – 100%**	
Field Visit, Plan, Discussion	10 Marks	• Implementation strategies of the project. • The last report should be signed by SCR Coordinator, the HOD and Principal. • At last report should be evaluated by the SCR Coordinator. • Finally, the consolidated marks sheet should be sent to the CoE/ university and to be made available at LIC visit.
Commencement of activities and its progress	20 Marks	
Case study-based Assessment Individual performance with report	20 Marks	
Sector wise study & its consolidation 5*5 = 25	25 Marks	
Video based seminar for 10 minutes by each student at the end of semester with Report. **Activities 1 to 5, 5*5 = 25**	25 Marks	

Total marks for the course in each semester	**100 Marks**	
For each activity, 20 marks CIE will be evaluated for IA marks at the end of semester, Report and assessment copy should be made available in the department.		
Students should present the progress of the activities as per the schedule in the prescribed practical session in the field. There should be positive progress in the vertical order for the benefit of society in general through activities.		

Guideline for Assessment Process:

Continuous Internal Evaluation (CIE)

After completion of the course, the student shall prepare, with daily diary as reference, a comprehensive report in consultation with the mentor/s to indicate what he has observed and learned in the social connect period. The report should be signed by the mentor. The report shall be evaluated on the basis of the following criteria and/or other relevant criteria pertaining to the activity completed. Marks allotted for the diary are out of 50. Planning and scheduling the social connect Information/Data collected during the social connect Analysis of the information/data and report writing Considering all above points allotting the marks as mentioned below:

Excellent : 80 to 100
Good : 60 to 79
Satisfactory : 40 to 59
Unsatisfactory and fail : <39

SPECIAL NOTE: NO SEMESTER END EXAM (SEE) – COMPLETELY PRACTICAL AND ACTIVITIES-BASED EVALUATION

ACTIVITIES

Jamming session, open mic, and poetry: Platform to connect to others. Share the stories with others. Share the experience of Social Connect. Exhibit the talent like playing instruments, singing, one-act play, art-painting, and fine art.

PEDAGOGY

The pedagogy will include interactive lectures, inspiring guest talks, field visits, social immersion, and a course project. Applying and synthesizing information from these sources to define the social problem to address and take up the solution as the course project, with your group. Social immersion with NGOs/social sections will be a key part of the course. Will all lead to the course project that will address the needs of the social sector?

COURSE TOPICS

The course will introduce social context and various players in the social space, and present approaches to discovering and understanding social needs. Social immersion and inspiring

conversion will culminate in developing an actual, idea for problem-based intervention, based on an in-depth understanding of a key social problem.

DURATION

A total of 30 - 35 hours engagement per semester is required for the 3rd semester of the B.E./B.Tech. program. The students will be divided into groups. Each group will be handled by faculty mentors. Faculty mentors will design the activities (particularly Jamming sessions open mic, and poetry) Faculty mentors have to design the evaluation system as per VTU guidelines of scheme & syllabus.

PEDAGOGY – GUIDELINES

It may differ depending on local resources available for the study as well as environment and climatic differences, location and time of execution.

Sl. No	Topic	Group size	Location	Activity Execution	Reporting	Evaluation of the Topic
1.	**Plantation and adoption of a tree:**	May be individual or team of 5	Farmers land/ parks / Villages / roadside/ community area / College campus etc…..	Site selection /proper consultation/Continuous monitoring/ Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty
2.	**Heritage walk and crafts corner:**	May be individual or team of 5	Temples / monumental places / Villages/ City Areas / Grama panchayat/ public associations/ Government Schemes officers/ campus etc…..	Site selection /proper consultation/ Continuous monitoring/ Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty
3.	**Organic farming and waste management:**	May be individual or team of 5	Farmers land / parks / Villages visits / roadside/ community area / College campus etc…..	Group selection / proper consultation / Continuous monitoring / Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty
4.	**Water conservation:** & conservation techniques	May be individual or team of 5	Villages/ City Areas / Grama panchayat/ public associations/Government Schemes officers / campus etc…..	site selection / proper consultation/Continuous monitoring/ Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty
5.	**Food walk: Practices in society**	May be individual or team of 5	Villages/ City Areas / Grama panchayat/ public associations/Government Schemes officers/ campus etc…..	Group selection / proper consultation / Continuous monitoring / Information board	Report should be submitted by individual to the concerned evaluation authority	Evaluation as per the rubrics Of scheme and syllabus by Faculty

Plan of Action (Execution of Activities)

Sl. NO	Practice Session Description
1	Lecture session in field to start activities
2	Students Presentation on Ideas
3	Execution of Activity - Plantation and adoption of a tree
4	Execution of Activity - Heritage walk and crafts corner
5	Execution of Activity - Organic farming and waste management
6	Execution of Activity - Water conservation
7	Execution of Activity - Food walk
8	Case study based Assessment, Individual performance
9	Sector/ Team wise study and its consolidation
10	Video based seminar for 10 minutes by each student At the end of semester with Report.

- Each student should do activities according to the scheme and syllabus.
- At the end of semester student performance has to be evaluated by the faculty for the assigned activity progress and its completion.
- At last consolidated report of all activities from 1st to 5th, compiled report should be submitted as per the instructions and scheme.

B.E. (Common to all Branches) Outcome Based Education(OBE) and Choice Based Credit System(CBCS) SEMESTER – III			
Course Title:	**FOREIGN LANGUAGE COURSE 1- GERMAN**		
Course Code:	**BFLC358A**	**CIE MARKS**	**50**
(L :T : P) + SL	**(1:0:0) + (15 Hours/ Sem)**	**SEE MARKS**	**50**
Credits:	**1**	**EXAM HOURS**	**02**

COURSE LEARNING OBJECTIVES:

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

1. To develop basic communication skills in German.
2. To introduce German grammar and sentence structure.
3. To enhance vocabulary in everyday and academic contexts.
4. To enable students to comprehend and respond in spoken and written German.
5. To build intercultural awareness through authentic material like short films and dialogues.

Module 1	**3 Hours**
Vocabulary for day to day activities, Time – official/inofficial, Possessive articles, Speaking about family, Time prepositions, Worksheets, Short Film Textbook 1: Chapter 5, 6	
Module 2	**3 Hours**
Talk about hobbies, Talking about plans, Trennbare verben/seperable verbs, Going to a restaurant, Worksheets, Short Film Textbook: Chapter 5, 6	
Module 3	**3 Hours**
Talking about jobs, Talking about the workplace, Helping verbs in Past tense , Attending events in your town, Worksheets, Short Film Textbook: Chapter 7, 8	
Module 4	**3 Hours**
Daily routine, Body parts, Common illnesses, Visiting the doctor, Healthy living, Worksheets, Short Film Textbook: Chapter 7, 8	
Module 5	**3 Hours**
Dativ Imperative, Small Talk…Worksheets, Short Film Textbook: Chapter 7, 8	

Course outcomes

At the end of the course the student will be able to:

CO1: Understand and apply basic German grammar and sentence structure.
CO2: Communicate using appropriate vocabulary and expressions in daily situations.

CO3: Fill out simple forms and conduct structured conversations.
CO4: Comprehend basic written and audio-visual content in German.
CO5: Recognize basic cultural and linguistic distinctions in German-speaking countries.

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Average of Two Internal Assessment Tests (MCQs) each conducted for 25 marks.
Alternate Assessment (B)	20	-	Any two Assessment methods as per 24RNBE4.2, scaled to 25 marks.
Total CIE (A+B)	50	20	The sum of two tests, two assignments, will be out of 100 marks and will be scaled down to 50 marks.
SEE (C)	50	18	Question papers having 50 MCQs with each question carrying 01 mark, and Examination duration is 2 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.			

Textbooks

1. Netzwerk A1 Neu Kursbuch
2. Netzwerk A1 Neu Ubungsbuch

Web links and Video Lectures (e-Resources)

- https://www.youtube.com/playlist?list=PLquImyRfMt6esqE6C4RdOeg0h0IBeYWri
- https://www.youtube.com/playlist?list=PLWO81tQUNYnX7laegkYvzQiMrSghMXYCZ
- https://www.youtube.com/watch?v=4-eDoThe6qo
- https://learngerman.dw.com/en/wer-spricht/l-56577891/e-57119877

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- Role Play
- Flipped classroom
- Debate
- Films
- Quiz
- Games
- Handouts

B.E. (Common to all Branches) Outcome Based Education(OBE) and Choice Based Credit System(CBCS) SEMESTER – III			
Course Title:	**FOREIGN LANGUAGE COURSE 1- SPANISH**		
Course Code:	**BFLC358B**	**CIE MARKS**	**50**
(L :T : P) + SL	**(1:0:0) + (15 Hours/ Sem)**	**SEE MARKS**	**50**
Credits:	**1**	**EXAM HOURS**	**02**

COURSE LEARNING OBJECTIVES:

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

1. To introduce learners to the basics of Spanish phonetics and vocabulary.
2. To familiarize learners with basic sentence elements such as articles, pronouns, and regular verbs.
3. To promote simple self-expression and comprehension in everyday scenarios.
4. To prepare the learners to introduce and describe a person.

Module 1	**3 Hours**

Introduction and Phonetics:

- Spanish alphabet and sounds.
- Numbers (0–50), days and months.
- Greetings and introducing yourself.

Textbook: Chapter 0 and Chapter 1 – Lesson 2

Module 2	**3 Hours**

Nouns and Articles

- Gender and number of nouns.
- Definite and indefinite articles.
- Simple sentence structure with nouns.

Textbook: Chapter 2 – Lesson 3

Module 3	**3 Hours**

Subject Pronouns and Regular Verbs

- Subject pronouns (yo, tú, él, etc.).
- Present tense of regular AR/ER/IR verbs.
- Making positive and negative sentences.

Textbook: Chapter 2 – Lesson 4

Module 4	**3 Hours**

Ser vs Estar and Adjectives

- Usage of 'ser' and 'estar'.
- Descriptive adjectives and agreement.
- Describing people and places.

Textbook: Chapter 6 – Lesson 11

Module 5	**3 Hours**

Demonstratives, Possessives, and Family

- Demonstrative adjectives (este, esa, etc.).
- Possessive adjectives (mí, tu, su, etc.).
- Talking about family and relationships.

Textbook: Chapter 4 – Lesson 7 & Lesson 8; Chapter 6 – Lesson 12

Course outcomes

At the end of the course the student will be able to:

CO1: Recognize and pronounce basic Spanish phonemes and vocabulary.
CO2: Use articles, nouns, subject pronouns and regular verbs in simple sentences.
CO3: Construct and understand basic descriptive sentences using 'ser' and 'estar'.
CO4: Identify and describe relationships using demonstratives and possessives.

Assessment Details (both CIE and SEE)

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Average of Two Internal Assessment Tests (MCQs) each conducted for 25 marks.
Alternate Assessment (B)	20	-	Any two Assessment methods as per 24RNBE4.2, scaled to 25 marks.
Total CIE (A+B)	50	20	The sum of two tests, two assignments, will be out of 100 marks and will be scaled down to 50 marks.
SEE (C)	50	18	Question papers having 50 MCQs with each question carrying 01 mark, and Examination duration is 2 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.			

Textbook:

1. Chicos Chicas 1
2. Resources and Exercises from Aventura Hispana

Weblinks:

- **YouTube channels:** Butterfly Spanish, Dreaming Spanish

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- Role Play
- Debate
- Short videos
- Games
- Handouts

B.E. (Computer Science and Engineering) Outcome Based Education (OBE) and Choice Based Credit System (CBCS) SEMESTER – III			
Course Title:	**AI Tools and Applications**		
Course Code:	**BCIL358C**	**CIE MARKS**	**50**
(L: T: P) +SL	**0:0:2**	**SEE MARKS**	**50**
Credits:	**1**	**EXAM HOURS**	**03**

COURSE LEARNING OBJECTIVES:

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

1. Use Generative AI tools like ChatGPT, Canva, and Gamma. app for accelerating the creation of content, interactive simulations and assessments, to transform the raw material into structured, ready-to-use assets viz., content documents, presentations or visual handouts effortlessly.
2. Understand Precision Prompting: Scholarly Question Design for Reliable AI Outputs.
3. Use practical methods for leveraging Artificial Intelligence (AI) to enhance the productivity and effectiveness of academic research—specifically in ideation, literature review, and visualization of research connections.
4. Craft Adaptive Architectures for Personalised Learning Pathways using AI tools.

Program 1

Create high quality, structured course content Using Generative AI tools (Use ChatGPT).

Example: Create a lecture on looping statements in Python, World War II, etc.

Program 2

Demonstrate the Canvas mode in ChatGPT

Example 1: Show the Interactive Editing of Documents using prompt-based modifications. Apply a prompt like: "Make this shorter", "Add an analogy here", "Convert this into bullet points".

Example 2: Write a paragraph on if-else statements, select it, and prompt: "Explain this with a real-life analogy for beginners."

Program 3

Demonstrate the usage of Notion (knowledge management platform) which is a smart, flexible Google Doc with extra features such as Nested pages, Toggle sections, Visual blocks (tables, images, embeds), Easy rearrangement.

Example: Create or demonstrate how to:

- Paste GPT-generated content into Notion
- Break it into pages or collapsible sections
- Prepare content for sharing or lesson delivery

Program 4

Demonstrate the use of Canva Docs and Canva Magic Tools.

Example: Demonstrate how to take the AI-generated lesson (on looping Statements in Python) and copy it directly into Canva Docs, which is Canva's version of a dynamic

document editor. Paste GPT/Notion Content into Canva Docs using Canva Magic Tools. Create a visually enhanced handout or lesson packet that students can download, read on mobile, or print.

After generating the topic breakdown with ChatGPT:

- Use Canva Docs to format each cause as a section
- Add relevant photos from Canva Elements
- Include a "Key Takeaway" block after each section
- Attach a short 2-question quiz at the end

Program 5

Demonstrate the use of Magic Suite-Magic Eraser/Replace, Magic transform, etc.

Example:

a. Demonstrate the Magic Write feature. Say, Right-click on a paragraph, then choose actions like: "Rewrite this", "Make it more concise", "Fix grammar", "Expand this with an example", "Custom prompt" (e.g., "Explain this in simple language")
b. Demonstrate the Magic Eraser / Replace feature which is Primarily used in images. Say, Erase unwanted objects (e.g., a watermark or irrelevant icon), Replace an item in an image (e.g., change a laptop to a chalkboard)
c. Demonstrate the Magic Transform feature which can be used to instantly convert the above created document into multiple formats.

Program 6

Demonstrate the use of Gamma.app-AI-Powered presentation builder.

Show the following Customizations for the document created in above examples.

Feature	**Example**
Slide Title/Text	Edit headings or simplify slide copy
Image Replacement	Use AI to regenerate more suitable visuals
Content Rewrite	Prompt: "Make this slide simpler for high school students"
Layout Editing	Rearrange card structure or duplicate/delete cards

Program 7

Generate Questions from Content (CTRL Method). CTRL is a powerful method for precision prompting.

Example: Python Conditional Statements

- Prompt Example 1: Create 3 Multiple Choice Questions (MCQs) on Conditional Statements in Python for college students who are beginners. Keep the tone formal and straightforward.
- Prompt Example 2: Generate 2 Match-the-columns questions on Python Conditional Statements. Keep the questions beginner-friendly and formal.

- Demo Examples:
- Using PDF as Content Source:
  - ✓ I've uploaded a PDF on "Introduction to Sociology" (provide link). Generate 5 MCQs for undergraduate students. Maintain academic tone, moderate complexity.
- Using Web Link:
  - ✓ Use the content from the following web link (provide actual URL here) and create 3 MCQs on Photosynthesis suitable for high-school biology students.

Program 8

Demonstrate the Parsons Method for Interactive Problem Statements.

Parson's Method enhances problem-solving skills by giving students code fragments or sentences they must put in the correct logical order, proven to improve conceptual understanding.

Live Demonstration Prompt:

Generate a Parson's Problem using Python conditional statements for beginner-level students. Provide shuffled code blocks for a simple "if-else" structure.

Program 9

Demonstrate the usage of AI Tools for Creating and Delivering Tests.

Tools available are: Quizizz AI, ClassMarker, Formative, Google Forms + AI (Form Presenter)

Demonstrate a Use case of AI Tools for Creating and Delivering Tests.

Once questions are created (Q7), automate their conversion into online tests for ease of deployment and analysis. Use Quizizz AI for Automated MCQs, grading and analytics.

Program 10

Demonstrate the use of AI-Driven Research Ideation (ChatGPT + Search)

Live Demonstration Example:

Example Prompt given to ChatGPT:

Suggest 3 novel research angles on the topic:

'Impact of Social Media on Mental Health' relevant to Psychology researchers.

Program 11

Deploy Elicit (elicit.com) for Rapid Literature Discovery.

To overcome manual search limitations, use Elicit, an AI-driven tool specialized in academic literature discovery.

Example: Topic provided: "What is the impact of mindfulness techniques on anxiety among university students?"

Analyzing Search Results:

Elicit automatically generates a structured table of scholarly papers directly related to your query. It quickly provides: Paper titles, Abstracts (automatically summarized), Methodologies, Participants/sample sizes, Year published, citations count, and study types.

The student should Customize Table Columns: say,

Click "Columns" →Select additional details to view: Population studied, Intervantion details, Outcomes measured, Research methods Show the Advanced Filtering techniques: Say • Filter results by: Publication date, Study type (e.g., randomized control trials, observational studies), Citations count (to identify influential papers)
Program 12
Using Consensus for Quick Expert Insights (consensus.app). The Consensus AI tool summarizes academic research consensus clearly and visually. Example: Enter your research query.: "Does social media use negatively impact teenagers' mental health?" Understand the Results: "Consensus Meter" summarizing expert agreement, Conduct In-depth Exploration of each summarized statement, do Advanced Filtering (Narrow down by year, relevance, study type, or agreement level.
Program 13
Visualizing Research Connections with Litmaps (app.litmaps.com) Litmaps, a visualization tool that simplifies understanding complex academic citation networks visually Example: Enter your core research paper or DOI (Digital Object Identifier). Explore the Visual Citation Map (Connected papers represented visually, indicating citation relationships clearly. Papers connected via arrows showing who cites whom). Interactivity and Exploration: Click on any paper node to: Quickly see its abstract, authors, citations. Explore further connections expanding research context. Refine the Map: Filter the visualizations by: Year, Citations, Keywords, Publication type
Program 14
Explore how Generative AI can be applied in education to enable personalised learning experiences and support learners across various tasks. Use AI Tools such as Khanmigo, QuestionWell (For generating questions aligned to standards) or Diffit (For creating differentiated instructional materials). • Example Prompt 1: "Act as a backend software development interviewer. Ask questions about JavaScript, React, Node.js, SQL, AWS, and System Design. Provide feedback based on a rubric." • Example Prompt 2: "Act as a Python Developer. Create a learning plan for someone transitioning from another programming language. Ask clarification questions before proceeding." • Example Prompt 3: "Provide constructive feedback on the student's answer based on the rubric. Mention 2 strengths, 2 areas for improvement, and suggestions. Use a positive tone." Tools You Can Explore: Brisk (Chrome extension for Google Docs), MagicSchool.ai (AI assistant for teachers).

COURSE OUTCOMES:

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

CO1	**Apply** Generative AI tools such as ChatGPT, Canva, and Gamma.app to create structured content, simulations, and assessments.
CO2	**Transform** raw information into well-structured academic or professional outputs using AI tools like Canva Docs and Gamma.app.
CO3	**Design** precision prompts to generate reliable scholarly content and evaluation questions using advanced prompting strategies.
CO4	**Utilize** AI-powered platforms (e.g., Elicit, Consensus, Litmaps) to enhance productivity in research ideation, literature review, and visualization.
CO5	**Create** adaptive and personalized learning pathways using AI tools to support differentiated instruction and learner engagement.

ASSESSMENT DETAILS (Both CIE and SEE)

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.			

Textbook

1. Sandeep Surapaneni, Introduction to AI for Non-Tech People
https://www.scribd.com/document/855410054/AI-for-Beginners-by-Sandeep-Surapaneni

Reference Links:

- OpenAI Prompt Engineering Guide: https://platform.openai.com/docs/guides/prompt-engineering
- Effective Few-shot prompting examples : https://learnprompting.org/docs/basics/few_shot_prompting

- Parsons Problems Explained (Research-backed) : https://computinged.wordpress.com/2017/06/05/parsons-problems-have-same-learning-gains-as-writing-or-fixing-code-with-less-frustration-and-more-fun/
- Example Parsons Problem Online Tool : https://parsons.problemsolving.io/
- Edutopia on AI Writing Feedback : https://www.edutopia.org/article/ai-writing-feedback-students/#:~:text=As%20a%20writing%20teacher%20I,I%20like%20them%20a%20lot

Tool	Link
ChatGPT (Canvas)	https://chat.openai.com
Notion	https://notion.so
Canva Docs	https://www.canva.com
Gamma.app	https://gamma.app
Genially	https://www.genial.ly
Quizizz AI	https://quizizz.com
H5P	https://h5p.org
AI Tutor	https://chatgpt.com/share/68358dc1-0748-8006-a2bf-c50023cc280f
Khanmigo	https://www.youtube.com/watch?v=rnIgnS8Susg&t=119s
Elicit	https://elicit.com/
Consensus	https://consensus.app/
Litmaps	https://app.litmaps.com/

B.E. (Computer Science and Engineering) Outcome Based Education (OBE) and Choice Based Credit System (CBCS) SEMESTER – III			
Course Title:	**Green IT And Sustainability**		
Course Code:	**BCST358D**	**CIE MARKS**	**50**
(L: T: P) + SL	**0:2:0 + (15 Hours / Sem)**	**SEE MARKS**	**50**
Credits:	**1**	**EXAM HOURS**	**2**

COURSE LEARNING OBJECTIVES:

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

1. Understand challenges for Green ICT and the environmental impact.
2. Learn different aspects of ICT metrics and Sustainable Cloud Computing.
3. Explore the effects of software design on sustainability.

Module-1	**3 Hours**
Green ICT -History, Agenda, and Challenges Ahead: Introduction, Industrial Revolution, The Emergence of Information and Communication Technologies, The Agenda and Challenges Ahead	
Module-2	**3 Hours**
Emerging Technologies and Their Environmental Impact: Introduction, Number of Connected Devices, Increased, Functionality, Increased Number of Separate Functions, Increased Demand for Speed and Reliability, Obsolescence—The Problem of Backward Compatibility, The Other Side of the Balance Sheet, Videoconference as an Alternative to Business Travel, Dematerialization of Product Chain, Travel Advice/Road Traffic Control, Intelligent Energy Metering, Building Management Systems, Saving IT	
Module-3	**3 Hours**
Measurements and Sustainability: Introduction, ICT Technical Measures, Ecological Measures and Ethical Consideration, Systems Engineering for Designing Sustainable ICT-Based Architectures.	
Module-4	**3 Hours**
Sustainable Cloud Computing -Challenges in the Use of Cloud Computing as Green Technology, Cloud Computing and Sustainability, Sustainable Applications of Cloud Computing, Technologies Associated with Sustainable Cloud Computing, Future Prospects of Sustainable Cloud Computing, Reflections on Sustainable Cloud Computing Applications.	
Module-5	**3 Hours**
Sustainable Software Design: Overview and Scope, Evaluating Sustainability Effects, Sustainability and the Product Life Cycle, Direct Effects: Sustainability During Use, Runtime Energy Consumption Basics, Analyzing the Energy Consumption of an Application, Energy Consumption Reduction Using Physical Properties of Semiconductors, Optimizing the Energy Consumption of an Application: Compiler Techniques, Optimizing the Energy Consumption of an Application: Runtime Approaches.	

COURSE OUTCOMES:

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

CO1	Classify the challenges of Green ICT.
CO2	Relate emerging technologies to their environmental impact.
CO3	Apply ICT metrics to assess sustainability aspects.
CO4	Compare parameters of sustainable cloud computing.
CO5	Interpret the impact of software design on sustainability.

ASSESSMENT DETAILS (Both CIE and SEE)

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.
Alternate Assessment (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 MCQs with each question carrying 01 mark, and Exam duration shall before 2hours.**
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.			

Text Books

1. Green Information Technology - A Sustainable Approach, Mohammad Dastbaz Colin Pattinson, Babak Akhgar, Elsevier, 2015 Inc.
2. San Murugesan; G. R. Gangadharan, Harnessing Green IT: Principles and Practices, Wiley-IEEE Press.
3. Green IT: Technologies and Applications - Jae H. Kim & Myung J. Lee (2021). A scholarly compilation covering theoretical foundations and practical applications including energy-efficient hardware, sustainable software design, and green networking

Reference Books:

1. Green IT Strategies and Applications: Using Environmental Intelligence - Bhuvan Unhelkar. Offers a robust reference blending research, case studies, risk management, ISO standards (e.g., ISO 14000), and organizational transformation frameworks.
2. Sustainable IT Playbook for Technology Leaders - Niklas Sundberg et al. A modern, actionable guide for IT leaders to design and implement sustainable IT practices that align with business imperatives
3. Greening through IT - Tomlinson. Explores the theoretical, technological, and social dimensions of Green IT with real-world case studies like smart meters and mobile-enabled fishers.

B.E. (Computer Science and Engineering) Outcome Based Education (OBE) and Choice Based Credit System (CBCS) SEMESTER – III			
Course Title:	**Project Management with GIT**		
Course Code:	**BCSL358E**	**CIE MARKS**	**50**
(L: T: P) + SL	**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. To become familiar with the basic commands of Git.
2. To learn how to create and manage branches effectively.
3. To understand the concepts of collaboration using remote repositories.
4. To become proficient with version control commands.

Experiment 1. Setting Up and Basic Commands
Initialize a new Git repository in a directory. Create a new file and add it to the staging area and commit the changes with an appropriate commit message
Experiment 2. Creating and Managing Branches
Create a new branch named "feature-branch." Switch to the "master" branch. Merge the "feature-branch" into "master."
Experiment 3. Creating and Managing Branches
Write the commands to stash your changes, switch branches, and then apply the stashed changes.
Experiment 4. Collaboration and Remote Repositories
Clone a remote Git repository to your local machine.
Experiment 5. Collaboration and Remote Repositories
Fetch the latest changes from a remote repository and rebase your local branch onto the updated remote branch.
Experiment 6. Collaboration and Remote Repositories
Write the command to merge "feature-branch" into "master" while providing a custom commit message for the merge.
Experiment 7. Git Tags and Releases
Write the command to create a lightweight Git tag named "v1.0" for a commit in your local repository.
Experiment 8. Advanced Git Operations
Write the command to cherry-pick a range of commits from "source-branch" to the current branch
Experiment 9. Analysing and Changing Git History
Given a commit ID, how would you use Git to view the details of that specific commit, including the author, date, and commit message?
Experiment 10. Analysing and Changing Git History

Write the command to list all commits made by the author "JohnDoe" between "2023-01-01" and "2023-12-31."
Experiment 11. Analysing and Changing Git History
Write the command to display the last five commits in the repository's history.
Program 12. Analysing and Changing Git History
Write the command to undo the changes introduced by the commit with the ID "abc123"

COURSE OUTCOMES:

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

CO1	Use the basics commands related to git repository
CO2	Create and manage the branches
CO3	Apply commands related to Collaboration and Remote Repositories
CO4	Use the commands related to Git Tags, Releases and advanced git operations
CO5	Analyse and change the git history

ASSESSMENT DETAILS (Both CIE and SEE)

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Marks awarded for Weekly Conduction of Experiments and Submission of Laboratory records, scaled to 30 marks.
Alternate Assessment (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 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.			

Text Books

1. Pro Git - Scott Chacon, Ben Straub, Apress, Year: 2014 (2nd Edition)

2. Git Pocket Guide - Richard E. Silverman, O'Reilly Media, *Year*: 2013
3. Version Control with Git: Powerful Tools and Techniques for Collaborative Software Development - Jon Loeliger, Matthew McCullough, O'Reilly Media, *Year*: 2012 (2nd Edition)

Reference Books:

1. Beginning Git and GitHub: Version Control, Project Management, and Teamwork for the New Developer - Mariot Tsitoara, Apress (Springer Nature), Year: 2024.
2. Git: Project Management for Developers and DevOps Teams - Bernd Öggl, Michael Kofler, SAP PRESS, Year: 2022.
3. Learning Git - Michael Hartl, Ian Miell, O'Reilly Media, Year: 2022

B.E. (Common to All Branches) Outcome Based Education (OBE) and Choice Based Credit System (CBCS) SEMESTER – III			
Course Title:	**NSS-National Service Scheme**		
Course Code:	**BNSK359**	**CIE MARKS**	**100**
L : T : P: SL	**0:0:0:30**	**SEE MARKS**	**0**
Credits:	**0**	**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 the problem–solving.
2. Develop among them a sense of social & civic responsibility & utilize their knowledge in finding practical solutions to individual and community problems.
3. Develop competence required for group-living and sharing of responsibilities & gain skills in mobilizing community participation to acquire leadership qualities and democratic attitudes.

Module-1
Introduction and Basic Concepts of NSS Aims and Objective of NSS. Emblem, Moto, Flag Role and Responsibility of youth in various activities of NSS
Module-2
Youth Leadership Definition of Leadership and Important components (Social, Mental, Moral and Cultural) Qualities and Characteristics of good leaders Role of Youth leadership
Module-3
Community Work through NSS Community Work: Meaning and Nature Concept of regular activities undertaking of NSS
Module-4
NSS Activity-1
Module-5
NSS Activity-2

COURSE OUTCOMES:

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

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

ASSESSMENT DETAILS (Both CIE and SEE)

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
Evaluation based on Reports (A)	100	-	5 modules Reports, 5 modules X 10 M each = 50 M
Total CIE Theory (A)	**100**	**40**	
SEE (B)	**-**	**-**	**-**
CIE+SEE (A+B)	**100**	**40**	**Final marks shall be the sum of marks scored in Components A, B**
• The minimum marks to be secured in CIE is 40 out of allotted 50 marks with minimum 40 % scored in every component of CIE. • No SEE for this subject			

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.

Weblinks

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

B.E. (Common to all Branches) Additional Mathematics-I (Mandatory Learning Course) A bridge course for Lateral Entry Students			
Course Code:	**BMATDIP31**	CIE Marks	100
(L:T:P) + SL	(3:0:0) + (45 Hours/Sem)	Total Marks	100
Credits	0	Exam Hours	0

Course objectives:
The mandatory learning course **BMATDIP31** viz., **Additional Mathematics-I** aims to provide basic concepts of complex numbers, vector algebra, differential & integral calculus, vector differentiation and methods of solving first-order differential equations.

Module-1	**8 Hours**
Differential Calculus: Successive differentiation- problems. Taylor's & Maclaurin's series expansion - problems. Partial Differentiation: Euler's theorem (without Proof)-problems on first order derivatives only. Total derivatives-differentiation of composite functions. Jacobians of order two-Problems.	
Module-2	**8 Hours**
Complex Numbers: Definitions and properties. Modulus and amplitude of a complex number, Arganda's diagram, De-Moivre's theorem (without proof), Problems. **Vector Algebra:** Scalar and vectors. Addition, subtraction and multiplication of vectors-Dot and Cross products, problems. Scalar triple product, Problems.	
Module-3	**8 Hours**
Vector Differentiation: Differentiation of vector functions. Velocity and acceleration of a particle moving on a space curve. Scalar and vector point functions. Gradient, Divergence, Curl-simple problems. Solenoidal and irrotational vector fields-Problems.	
Module-4	**8 Hours**
Integral Calculus: Review of elementary integral calculus. Reduction formulae for $\sin^n$, $\cos^n x$, $\sin^n x \cos^n x$ (without proof) and evaluation of these with standard limits-problems. Double and triple integrals-Simple problems.	
Module-5	**8 Hours**
Ordinary Differential Equations (ODEs): Introduction-solutions of first order and first-degree differential equations: Variable separable method, Homogeneous differential equations, linear differential equations. Exact differential equations.	

Course outcomes: *At the end of the course the student will be able to*:

CO1	Use derivatives and partial derivatives to calculate the rate of change of multivariate functions.
CO2	Apply concepts of complex numbers and vector algebra to analyse the problems arising in a related area.
CO3	Analyse position, velocity and acceleration in two and three dimensions of vector-valued functions.
CO4	Learn techniques of integration including the evaluation of double and triple integrals.
CO5	Identify and solve first-order ordinary differential equations.

Assessment Details (CIE)

The weightage of Continuous Internal Evaluation (CIE) is 100%. The minimum passing mark for the CIE is 40% of the maximum marks (100). A student shall be deemed to have satisfied the academic requirements if the student secures not less than 40% (40 Marks out of 100) in the CIE.

Continuous Internal Evaluation:

- Three Unit Tests each of **20 Marks (duration 01 hour)**
  1. The first test at the end of the 5th week of the semester
  2. The second test at the end of the 10th week of the semester
  3. Third test at the end of the 15th week of the semester Two assignments each of **20 Marks**
  4. First assignment at the end of the 4th week of the semester
  5. Second assignment at the end of the 9th week of the semester
- The sum of three tests and two assignments will be out of 100 marks
- The student shall secure a minimum of 40% of marks of the course to qualify and become eligible for the award of a degree.

Textbook

1. Higher Engineering Mathematics: B. S. Grewal, Khanna Publishers, New Delhi, 43rd Ed., 2015.

Reference Books

1. Higher Engineering Mathematics: V. Ramana, McGraw-Hill Education, 11th Ed.
2. Engineering Mathematics: Srimanta Pal & Subodh C. Bhunia, Oxford University Press, 3rd Reprint, 2016.
3. A textbook of Engineering Mathematics: N.P Bali and Manish Goyal, Laxmi Publications, Latest edition.

4. Higher Engineering Mathematics: H.K. Dass and Er. Rajnish Verma, S. Chand Publication (2014).

Web links and Video Lectures (e-Resources):

- http://.ac.in/courses.php?disciplineID=111
- http://www.class-central.com/subject/math(MOOCs)
- http://academicearth.org/
- VTU EDUSAT PROGRAMME-20

Activity-Based Learning (Suggested Activities in Class)/ Practical Based learning

- Quiz
- Group assignment
- Seminars

IV SEMESTER SUBJECTS

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

Course Title:	**Analysis and Design of Algorithms**		
Course Code:	**BCS401**	**CIE MARKS**	**50**
(L: T: P) + SL	**(3:2:0) + (45 Hours/Sem)**	**SEE MARKS**	**50**
Credits:	**4**	**EXAM HOURS**	**03**

COURSE LEARNING OBJECTIVES:

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

1. Learn the methods for analyzing algorithms and evaluating their performance.
2. Demonstrate the efficiency of algorithms using asymptotic notations.
3. Solve problems using various algorithm design methods, including brute force, greedy, divide and conquer, decrease and conquer, transform and conquer, dynamic programming, backtracking, and branch and bound.
4. Learn the concepts of P and NP complexity classes.

Module-1 **9 Hours**

Fundamentals: What is an Algorithm? Fundamentals of Algorithmic Problem Solving, Important Problem types, fundamentals of the analysis of Algorithm Efficiency: Analysis Framework, Asymptotic Notations and Basic Efficiency Classes, Mathematical Analysis of Non-Recursive Algorithms, Mathematical Analysis of Recursive Algorithms.
Problem Complexity: Limitations of Algorithm Power, Non- Deterministic Algorithms, P, NP, NP Complete, and NP-Hard classes.

Textbook 1: Chapter-1(1.1-1.3), Chapter-2(2.1-2.4), Chapter-11(11.1-11.3)

Module-2 **9 Hours**

Brute-Force and Exhaustive search: Selection Sort, Bubble Sort, Sequential Search and String Matching, Assignment Problem, Knapsack Problem, Travelling Salesman Problem
Decrease-and-Conquer: Insertion sort, Topological Sorting
Divide-and-Conquer: Merge Sort, Quicksort, Max Min Algorithm, Strassen's Matrix Multiplication.

Textbook 1: Chapter-3 (3.1-3.2,3.4) Chapter-4 (4.1-4.2) Chapter-5 (5.1-5.2,5.4)
Textbook 2: Chapter-3 (3.3)

Module-3 **9 Hours**

Greedy Technique: Prim's Algorithm, Kruskals Algorithm, Dijkstra's Algorithm, Huffman Trees, Greedy Knapsack
Transform and Conquer: Balanced Search Trees (AVL trees), Heapsort.

Textbook 1: Chapter-9 (9.1-9.4) Chapter-6 (6.3-6.4) Text book 2: Chapter-4 (4.2)

Module-4 **9 Hours**

Dynamic Programming: Multistage Graph, The Knapsack Problem and Memory functions, Warshall's Algorithm, Floyd's Algorithms. **Space and Time Trade-offs:** Sorting by Counting (Comparison &Distributed), Input Enhancement in String Matching: Horspool's algorithm. **Textbook 1: Chapter-8 (8.2,8.4) Chapter-7 (7.1-7.2)**
Module-5 **9 Hours**
Backtracking: n-Queens Problem, Hamiltonian Circuit problem, Subset-Sum Problem. **Branch-and-Bound:** Assignment Problem, Knapsack Problem, Traveling Salesman Problem. **Decision Trees:** Decision Trees for Sorting. **Textbook 1: Chapter-12 (12.1-12.2) Chapter-11 (11.2)**

COURSE OUTCOMES:

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

CO1	Apply fundamental concepts of algorithmic problem solving, analyse algorithm efficiency using asymptotic notations, and differentiate problem complexity classes like P, NP, NP-Hard Complete, and NP-Hard.
CO2	Analyse brute-force, decrease-and-conquer, and divide-and-conquer algorithms to solve basic sorting, searching, and matrix problems.
CO3	Apply greedy techniques and transform-and-conquer strategies to solve real-world problems such as minimum spanning trees, shortest paths, and efficient sorting.
CO4	Solve optimization problems using dynamic programming and understand space-time trade-offs in algorithm design with string matching and counting techniques.
CO5	Apply backtracking and branch-and-bound strategies to constraint satisfaction and combinatorial problems and understand the use of decision trees in analysing algorithm complexity.

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.**
SEE (D)	**50**	**18**	**SEE is a theory exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall before 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. Anany Levitin, "Introduction to the Design and Analysis of Algorithms", 3rd Edition, Pearson 2014.
2. Ellis Horowitz, Satraj Sahni, Rajasekharam, "Fundamentals of Computer Algorithms", 2nd University Press Pvt. Ltd 2009.

Reference Books

1. Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, Clifford Stein "Introduction to Algorithms", 3rd MIT Press 2009.
2. Padma Reddy, "Analysis and Design of Algorithms",1st Sri Nandi Publications 2009.

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

Course Title:	**Database Management Systems**		
Course Code:	**BCS402**	**CIE MARKS**	**50**
(L : T : P) + SL	**(3:0:2) + (45 Hours/Sem)**	**SEE MARKS**	**50**
Credits:	**4**	**EXAM HOURS**	**03**

COURSE LEARNING OBJECTIVES:

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

1. Understand the fundamental concepts, architecture, and advantages of Database Management Systems and their role in data handling.
2. Design and model database schemas using Entity-Relationship (ER) diagrams and map conceptual designs to relational models.
3. Write efficient SQL queries for data definition, manipulation, and retrieval, and understand NoSQL databases and their applications.
4. Apply normalization techniques to design well-structured relational database schemas and eliminate redundancy.
5. Comprehend transaction processing, concurrency control, and recovery techniques to ensure data integrity and reliability in database systems.

Module – 1 **9 Hours**

Introduction to Databases: Introduction, Characteristics of database approach, Advantages of using the DBMS approach, History of database applications.

Overview of Database Languages and Architectures: Data Models, Schemas, and Instances. Three schema architecture and data independence.

Conceptual Data Modelling using Entities and Relationships: Entity types, Entity sets and structural constraints, Weak entity types, ER Diagrams, Naming Conventions, and Design Issues.

Textbook 1: Ch 1.1 to 1.7, 2.1 to 2.2, 2.5 to 2.6, 3.1 to 3.7

Module – 2 **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.

Relational Algebra: Unary and Binary relational operations, additional relational operations (aggregate, grouping, etc.).

Textbook 1: Ch 9.1; Ch 5.1 to 5.3, Ch 8.1 to 8.4

Module – 3 **9 Hours**

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. More complex SQL retrieval queries, Specifying constraints as assertions and action triggers, Views in SQL. **NOSQL Databases and Big Data Storage Systems:** Introduction to NOSQL Systems, The CAP Theorem, Document-Based NOSQL Systems and MongoDB. Textbook 1: Ch 6.1 to 6.5, Ch 7.1 to 7.3, Ch 24.1 to 24.3
Module – 4 **9 Hours**
Normalization: Database Design Theory - Introduction to Normalization using Functional and Multivalued Dependencies: Informal design guidelines for relation schema, Functional Dependencies, Normal Forms based on Primary Keys, Second and Third Normal Forms, Boyce-Codd Normal Form. **Relational Database Design Algorithms**: Functional Dependencies: Inference Rules, Equivalence, and Minimal Cover, Properties of Relational Decompositions, Algorithms for Relational Database Schema Design, About Nulls, Dangling Tuples. Textbook 1: Ch 14.1 to 14.5, Ch 15.1 – 15.4
Module – 5 **9 Hours**
Transaction Processing: Introduction to Transaction Processing, Transaction and System concepts, Desirable properties of Transactions, Characterizing schedules based on recoverability, Characterizing schedules based on Serializability, Transaction support in SQL. **Concurrency Control in Databases:** Two-phase locking techniques for Concurrency control, Concurrency control based on Timestamp ordering. **Database Recovery Techniques**: Recovery Concepts, NO-UNDO/REDO Recovery Based on Deferred Update, Recovery Techniques Based on Immediate Update. Textbook 1: Ch 20.1 to 20.6, Ch 21.1 to 21.2, Ch 22.1 – 22.5, 22.7

PRACTICAL COMPONENT OF IPCC

Sl. No.	EXPERIMENTS
	PART A
1	Introduction of concept of rows and columns with attribute names using MS-Excel File or Spreadsheet with multiple sheets representing multiple tables.
2	Introduction of simple databases like SQL Lite / Oracle 11g / MySQL or online SQL interfaces such as https://livesql.oracle.com/ with CREATE, ALTER, DROP, INSERT, DELETE, UPDATE, GRANT, REVOKE,… SQL statements

3	Introduction to ER – Diagram and Schema writing with all the concepts like types of attributes, entity types, participation ratio, cardinality ratio, structural constraints, etc., Case studies on databases such as Banking, Airline, Employee, Library to be discussed in detail.
4	Installation of an Open-Source NoSQL Database MongoDB & perform basic CRUD (Create, Read, Update & Delete) operations. Execute MongoDB basic Queries using CRUD operations. https://www.mongodb.com/try/download/community
5	Consider the following schema for a Library Database: BOOK (Book_id, Title, Publisher_Name, Pub_Year) BOOK_AUTHORS (Book_id, Author_Name) PUBLISHER (Name, Address, Phone) BOOK_COPIES (Book_id, Branch_id, No-of_Copies) BOOK_LENDING (Book_id, Branch_id, Card_No, Date_Out, Due_Date) LIBRARY_BRANCH (Branch_id, Branch_Name, Address) BORROWER(Card_no, Name, Address, Phone) Write SQL queries to a. Retrieve details of all books in the library – id, title, name of publisher, authors, number of copies in each branch, etc. b. Get the particulars of borrowers who have borrowed more than 3 books, but from Jan 2017 to Jun 2017. c. Delete a book in BOOK table. Update the contents of other tables to reflect this data manipulation operation. d. Partition the BOOK table based on year of publication. Demonstrate its working with a simple query. e. Create a view of all books and its number of copies that are currently available in the Library.
6	Consider the following schema for Order Database: SALESMAN (Salesman_id, Name, City, Commission) CUSTOMER (Customer_id, Cust_Name, City, Grade, Salesman_id) ORDERS (Ord_No, Purchase_Amt, Ord_Date, Customer_id, Salesman_id) Write SQL queries to a. Count the customers with grades above Bangalore's average. b. Find the name and numbers of all salesmen who had more than one customer. c. List all the salesman and indicate those who have and don't have customers in their cities (Use UNION operation.) d. Create a view that finds the salesman who has the customer with the highest order of a day. e. Demonstrate the DELETE operation by removing salesman with id 1000. All his orders must also be deleted.

7	Consider the schema for Movie Database: ACTOR (Act_id, Act_Name, Act_Gender) DIRECTOR (Dir_id, Dir_Name, Dir_Phone) MOVIES (Mov_id, Mov_Title, Mov_Year, Mov_Lang, Dir_id) MOVIE_CAST (Act_id, Mov_id, Role) RATING (Mov_id, Rev_Stars) Write SQL queries to a. List the titles of all movies directed by 'Hitchcock'. b. Find the movie names where one or more actors acted in two or more movies. c. List all actors who acted in a movie before 2000 and also in a movie after 2015 (use JOIN operation). d. Find the title of movies and number of stars for each movie that has at least one rating and find the highest number of stars that movie received. Sort the result by movie title. e. Update rating of all movies directed by 'Steven Spielberg' to 5
8	Consider the schema for College Database: STUDENT (USN, SName, Address, Phone, Gender) SEMSEC (SSID, Sem, Sec) CLASS (USN, SSID) SUBJECT (Subcode, Title, Sem, Credits) IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3, FinalIA) Write SQL queries to a. List all the student details studying in fourth semester 'C' section. b. Compute the total number of male and female students in each semester and in each section. c. Create a view of Test1 marks of student USN '1RN25CS100' in all subjects. d. Calculate the FinalIA (average of best two test marks) and update the corresponding table for all students. e. Categorize students based on the following criterion: If FinalIA = 17 to 20 then CAT = 'Outstanding' If FinalIA = 12 to 16 then CAT = 'Average' If FinalIA< 12 then CAT = 'Weak'. Give these details only for 8th semester A, B, and C section students.

9	Consider the schema for Company Database: EMPLOYEE (SSN, Name, Address, Sex, Salary, SuperSSN, DNo) DEPARTMENT (DNo, DName, MgrSSN, MgrStartDate) DLOCATION (DNo, DLoc) PROJECT (PNo, PName, PLocation, DNo) WORKS_ON (SSN, PNo, Hours) Write SQL queries to a. Make a list of all project numbers for projects that involve an employee whose last name is 'Scott', either as a worker or a s a manager of the department that controls the project. b. Show the resulting salaries if every employee working on the 'IoT' project is given a 10 percent raise. c. Find the sum of the salaries of all employees of the 'Accounts' department, as well as the maximum salary, the minimum salary, and the average salary in this department. d. Retrieve the name of each employee who works on all the projects controlled by department number 5 (use NOT EXISTS operator). e. For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than Rs. 6,00,000.
	PART B
10	**Mini Project -** Batch formation with a 2 – 3 students in a batch and topic finalization. **The following are to be included in the mini – project:** a. Design of database using ER Diagram / Schema Diagram b. Implementation of backend using both Structured and Unstructured databases. c. Front-end design using any scripting languages like JavaScript, PHP, Python etc. d. Must include an innovative component in the implementation of the project. e. Projects must be uploaded into GitHub repository for final submission with public access and all the documentation for demonstration. f. A mini project report with soft binding with formats as specified by the department to be submitted at the end after demonstration.

COURSE OUTCOMES:

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

CO1	Define the fundamentals of DBMS architecture, data models, and ER modelling concepts.
CO2	Construct ER diagrams into relational schemas and apply relational algebra operations.
CO3	Formulate SQL queries and **differentiate** between SQL and NoSQL databases.
CO4	Apply normalization techniques to design relational schemas up to BCNF.
CO5	**Analyse** transaction processing and **implement** concurrency control and recovery techniques.

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.**
SEE (D)	**50**	**18**	**SEE is a theory exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall before 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. Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.

Reference Books

1. Database Management Systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill.
2. Silberschatz Korth and Sudharshan, Database System Concepts, 6th Edition, Mc-Graw Hill, 2013.

Weblinks:

1. https://www.youtube.com/@databasemanagementsystem-i4463 (NPTEL)
2. https://www.youtube.com/playlist?list=PLxCzCOWd7aiFAN6I8CuViBuCdJgiOkT2Y
3. https://www.youtube.com/watch?v=c5HAwKX-suM&ab_channel=SCALER
4. https://www.geeksforgeeks.org/dbms/dbms/

Self – Study topics:

Module 3: (Ch 24.4 – 24.6)
NOSQL Key-Value Stores, Column-Based or Wide Column NOSQL Systems, NOSQL Graph Databases and Neo4j

Module 4: (Ch 14.6 – 14.7)
Multivalued Dependency and Fourth Normal Form, Join Dependencies and Fifth Normal Form.

Module 5: (21.3 – 21.5)
Multiversion Concurrency control techniques, Validation Concurrency control techniques, Granularity of Data items and Multiple Granularity Locking.

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning Mini Project:

1. https://www.hackerrank.com/domains/sql - to explore the problem solving
2. Case Study based flip class

B.E. (Computer Science and Engineering) Outcome Based Education (OBE) and Choice Based Credit System (CBCS) SEMESTER – IV			
Course Title:	**Microcontrollers and IoT**		
Course Code:	**BCS403**	**CIE MARKS**	**50**
(L: T: P) + SL	**(3:0:2) + (45 Hours / Sem)**	**SEE MARKS**	**50**
Credits:	**4**	**EXAM HOURS**	**03**

COURSE LEARNING OBJECTIVES:

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

1. Understand the basics of ARM architecture, registers, exceptions, and interrupts.
2. Use ARM instructions for data processing, memory access, and software interrupts.
3. Write efficient C programs and understand how interrupts are handled.
4. Learn about sensors, actuators, and IoT communication protocols.
5. Build simple IoT applications using Arduino Uno and communication modules.

Module-1	**9 Hours**
ARM Processor Fundamentals- Registers, Current Program Status Register, Exceptions, Interrupts, Vector Table. **Introduction to the ARM Instruction Set-** Data Processing Instructions, Branch Instructions. **Text book 1: Chapter-2(2.1-2.2,2.4),Chapter-3(3.1-3.2)**	
Module-2	**9 Hours**
Load-Store Instructions, Software Interrupt Instruction. Program Status Register Instructions. **Text book 1: Chapter-3(3.3-3.5(Excluding 3.5.1,3.5.2))**	
Module-3	**9 Hours**
Efficient C Programming Overview of C Compilers and Optimization, Basic C Data Types, C Looping Structures. Exception and Interrupt Handling, Exception Handling, Interrupts. **Textbook 1: Chapter 5(5.1-5.3),Chapter-9(9.1-9.2)**	
Module-4	**9 Hours**
What is IoT? **Transducers, Sensors and Actuators-** Defining Transducers, Sensors and Actuators, Introduction to transducers, Introduction to Sensors. **IoT Protocols-** Protocol Classification, MQTT, COAP, REST. **Textbook2: Chapter-1(1.1), Chapter-2 (2.1-2.2,2.3.1,2.3.2.1-2.3.2.2,2.4), Chapter-3 (3.1-3.2,3.6-3.7)**	
Module-5	**9 Hours**
Introduction to Arduino, Exploring Arduino Uno Learning Board, Installing the software Fundamentals of Arduino Programming, Introduction to Communications, Example Module on Arduino. **Textbook 2: Chapter-7(7.1-7.6)**	

PRACTICAL COMPONENT OF IPCC

Sl. No.	EXPERIMENTS
	PART A
1	Write an ALP to find the sum of the first 10 integer numbers.
2	Write an ALP to i)Multiply two 16-bit numbers. ii) Add two 32-bit numbers.
3	Write an ALP to find the factorial of a number.
4	Write an ALP to add an array of 16-bit numbers and store the 32-bit result in internal RAM.
5	Write an ALP to find the square of a number (1 to 10) using a look-up table.
6	Write an ALP to find the largest or smallest number in an array of 32 numbers.
7	Write an ALP to arrange a series of 32-bit numbers in ascending/descending order.
8	Write an ALP to count the number of ones and zeros in two consecutive memory locations.
9	Assembly Language Program (ALP).
	PART B
10	**Using any IoT Interfacing Boards** 1. Develop a program to blink 5 LEDs back and forth. 2. Develop a program to control a DC motor. 3. Develop a program to deploy smart streetlight system using LDR sensor. 4. Develop a program to classify dry and wet waste with the Moisture sensor (DHT22). 5. Develop a program to detect the gas leakage in the surrounding environment. 6. Develop a water level depth detection system using Ultrasonic sensor.
	Demo Experiments
11	1. Develop a program to setup a UART protocol and pass a string through the protocol. 2. Develop a program to demonstrate weather station readings.

COURSE OUTCOMES:

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

CO1	Interpret the ARM architecture, registers, exceptions, interrupts, and the vector table.
CO2	Apply ARM instructions for data processing, branching, and software interrupts.
CO3	Demonstrate efficient embedded C programming techniques and optimization strategies.
CO4	Illustrate the role of sensors and actuators in IoT systems, and evaluate various IoT communication protocols

CO5	Develop simple IoT-based embedded applications using MCUs and sensor modules.

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.**
SEE (D)	**50**	**18**	**SEE is a theory exam, conducted for 100 marks and scaled down to 50 marks. Exam duration shall before 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. N. Sloss, D. Symes, and C. Wright, "ARM System Developer's Guide", Amsterdam, Netherlands: Elsevier/Morgan Kaufmann, 2008.
2. K. G. Srinivasa, G. M. Siddesh, and R. Hanumantha Raju, "Internet of Things", New Delhi, India: Cengage India, 2018.

Reference Books:

1. Raghunandan, "Microcontroller (ARM) and Embedded System", New Delhi, India: Cengage Learning, 2019.
2. A. Bahga and V. Madisetti, "Internet of Things: A Hands-On Approach", 1e. Universities Press, 2014.

Weblinks:

1. https://onlinecourses.nptel.ac.in/noc20_cs15/preview
2. https://www.arm.com
3. https://en.wikipedia.org/wiki/ARM_architecture_family
4. https://developer.arm.com/documentation
5. https://developer.arm.com/documentation/dui0041/c/ARM-Compiler-Reference/C-and-C---implementation-details/Basic-data-types?utm_source=chatgpt.com
6. https://www.emqx.com/en/blog/iot-protocols-mqtt-coap-lwm2m?utm_source=chatgpt.com
7. https://www.fortinet.com/resources/cyberglossary/iot?utm_source=chatgpt.com
8. https://www.emqx.com/en/blog/iot-protocols-mqtt-coap-lwm2m?utm_source=chatgpt.com
9. https://www.arduino.cc

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning Mini Project:

- Flipped Classroom
- Role Play
- Group Discussion (GD)
- Model Building / Concept Mapping
- Peer Teaching / Student Seminars
- Project Based Learning

B.E. (Computer Science and Engineering) Outcome Based Education (OBE) and Choice Based Credit System (CBCS) SEMESTER – IV			
Course Title:	**Design and Analysis of Algorithms Laboratory**		
Course Code:	**BCSL404**	**CIE MARKS**	**50**
L: T: P	**0:0:2**	**SEE MARKS**	**50**
Credits:	**1**	**EXAM HOURS**	**03**

COURSE LEARNING OBJECTIVES:

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

1. Learn the methods for analysing algorithms and evaluating their performance to demonstrate the efficiency of algorithms using asymptotic notations.
2. Demonstrate the efficiency of algorithms using asymptotic notations.
3. Solve problems using various algorithm design methods, including brute force, greedy, divide and conquer, decrease and conquer, transform and conquer, dynamic programming, backtracking, and branch and bound.
4. Learn the concepts of P and NP complexity classes.

Program 1. Selection Sort – Sorting Student IDs

Design and implement a C program to sort a list of student roll numbers from a university database using the Selection Sort algorithm.

Objective: Record and compare the time taken for sorting as the number of student records increases (n > 5000).

Application: Helps in managing ordered student data efficiently for attendance systems or result generation.

Program 2. Merge Sort – Organizing Medical Reports

Design and implement a C program to sort large sets of patient report IDs using Merge Sort**.**

Objective: Track execution time for datasets with n > 5000 and plot time vs data size.

Application: Efficient sorting of medical records in a hospital management system.

Program 3. Quick Sort – E-Commerce Order Sorting

Design and implement a C program to sort customer order IDs based on timestamps using Quick Sort.

Objective: Measure performance on data size > 5000 and plot performance graph.

Application: Fast processing of online orders in an e-commerce platform.

Program 4. Topological Sort – Task Scheduling in Project Management

Implement a C program to find the topological ordering of tasks in a project where some tasks depend on the completion of others.

Application: Automating task scheduling tools like Jira or Trello where dependencies exist.

Program 5. 0/1 Knapsack – Cargo Optimization

Implement a C program to solve the 0/1 Knapsack problem to maximize the value of items packed in a delivery truck with weight constraints.

Application: Logistic companies optimizing truck loads with limited capacity

Program 6. Greedy Knapsack – Stock Refill Strategy

Implement a C/C++ program to solve discrete and fractional knapsack problems using greedy methods to choose items for shelf restocking in a supermarket based on value-to-weight ratio. **Application:** Inventory management and optimization.
Program 7. Dijkstra's Algorithm – GPS Navigation System
Implement a C/C++ program to find shortest paths from a given location to all other locations in a city's road network. **Application:** GPS systems like Google Maps, for calculating the shortest routes.
Program 8a. Floyd's Algorithm – Travel Time Optimization Problem:
Implement a C/C++ program to find the shortest paths between all pairs of cities in a country using Floyd's algorithm. **Application:** Railway or airline scheduling and route planning.
Program 8b. Warshall's Algorithm – Webpage Reachability
Implement a C/C++ program to determine reachability between all webpages using Warshall's algorithm on a hyperlink graph. **Application:** Search engines identifying which pages link to others.
Program 9. Kruskal's Algorithm – Designing Network Layouts
Implement a C/C++ program to find the Minimum Cost Spanning Tree of a network using Kruskal's algorithm. **Application:** Laying cables (like fiber optic) for minimum cost in a connected campus or city.
Program 10. Prim's Algorithm – Power Grid Optimization
Implement a C/C++ program to build the minimum cost electricity network using Prim's algorithm. **Application:** Designing cost-effective electricity distribution networks.
Program 11. Subset Sum – Payment Combinations
Implement a C/C++ program to find all payment combinations using a set of coin or currency denominations that sum up to a required amount. **Application:** ATM cash dispensing or vending machines.
Program 12. N-Queens – Security Camera Placement
Implement a C/C++ program to solve the N-Queens problem to place security cameras in a museum grid such that no two cameras cover the same row, column, or diagonal. **Application:** Surveillance system optimization.

COURSE OUTCOMES:

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

CO1	Analyse and implement sorting algorithms such as Selection Sort, Merge Sort, and Quick Sort to solve domain-specific problems.
CO2	Apply graph-based algorithms such as Topological Sort, Dijkstra's, Floyd's, Warshall's, Kruskal's, and Prim's algorithms to solve real-time optimization and scheduling problems
CO3	Design and develop efficient solutions using Dynamic Programming techniques for combinatorial problems.

CO4	Solve resource allocation problems using Greedy algorithms and demonstrate their effectiveness in constrained environments.
CO5	Implement backtracking techniques to solve constraint satisfaction problems such as the N-Queens problem.

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.			

Weblinks:

1. Virtual Labs (CSE): http://cse01-iiith.vlabs.ac.in/

Activity Based Learning (Suggested Activities in Class)/ Practical Based Learning Mini Project:

1. Simulation of Algorithms
2. Leet Coding

B.E. (CSE and Allied Branches)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – IV

Course Title:	**Linear Algebra**		
Course Code:	**BCS405A**	**CIE MARKS**	**50**
(L: T : P) : SL	**(3:0:0) + (45 Hours / Sem)**	**SEE MARKS**	**50**
Credits:	**3**	**EXAM HOURS**	**03**

COURSE LEARNING OBJECTIVES:

1. To help students to understand System of linear equations and vector spaces.
2. To impart basics of Eigen Values and Eigen Vectors and their importance.
3. To facilitate students in applying inner product in Gram- Smidth process.
4. To provide some special operations of vector spaces and Matrices.

Module-1	**9 Hours**
The geometry of linear equations, existence and uniqueness, LU-decomposition and computational complexity, Vector Spaces and subspaces (Definitions only), The column space and null space of a given matrix, linearly independent, Bases and dimension, Null space and column space of a matrix, The four fundamental subspaces. **Textbook 2: Ch - 1.2, 1.5, 2.1, 2.3, 2.4**	
Module-2	**9 Hours**
Eigen values and Eigen vectors, Diagonalization, Eigen vectors and linear transformations, Complex Matrices. **Textbook 2: Ch - 5.1,5.2, 5.5**	
Module-3	**9 Hours**
Inner product, length and orthogonality, orthogonal sets and projections, orthonormal Basis, Gram-Schmidt process, QR-factorization, least square approximation, Rotation. **Textbook 1: Ch- 6.1, to 6.4, 6.7**	
Module-4	**9 Hours**
Diagonalization of symmetric matrices, quadratic forms and its classifications, constrained optimisation and Singular value decomposition, Application to Image Processing. **Textbook 1: Ch - 7.1 to 7.5**	
Module-5	**9 Hours**
The intersection, sum, Cartesian product and the Tensor product of two vector spaces, The Kronecker Product of Two Matrices, Jordan canonical form. **Textbook 2: Ch A – A.1 to A.5 and B**	

ASSESSMENT DETAILS (Both CIE and SEE)			
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.**			

COURSE OUTCOMES:

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

CO1	**Explain** the fundamental concepts of vector spaces and subspaces.
CO2	**Apply** the concept of linear independence, basis and dimension in solving problems.
CO3	**Perform computations** using linear transformations and their matrix representations.
CO4	**Analyze** eigenvalues and eigenvectors and their applications.
CO5	**Evaluate** inner product spaces and orthogonality principles.

Textbooks:

1. David C. Lay, Steven R. Lay, Judi J Mc. Donald: "Linear Algebra and its applications", Pearson Education, 4th Edition, 2017.
2. Gilbert Strang: "Linear Algebra and its applications", Brooks Cole, 4th edition, 2005.

Reference Books:

1. Richard Bronson & Gabriel B. Costa: "Linear Algebra: An Introduction", 2nd edition. Academic Press, 2014.
2. Seymour Lipschutz, Marc Lipso: "Theory and problems of linear algebra", Schaum's outline series - 6th edition, 2017, McGraw-Hill Education.
3. Marc Peter Deisennroth, A. Aldo Faisal, Cheng Soon Ong: "Mathematics for Machine learning", Cambridge University Press, 2020.

Web links and Video Lectures (e-Resources):

1. https://ocw.mit.edu/courses/mathematics/18-06sc-linear-algebra-fall 2011/index.htm
2. https://www.math.ucdavis.edu/~linear/linear.pdf
3. https://www.coursera.org/learn/linear-algebra-machine-learning
4. https://nptel.ac.in/syllabus/111106051/
5. http://nptel.ac.in/courses.php?disciplineID=111
6. http://www.class-central.com/subject/math(MOOCs)
7. http://academicearth.org/
8. VTU e-Shikshana Program
9. VTU EDUSAT Program.

B.E. (CSE and Allied Branches)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – IV

Course Title:	**Graph Theory**		
Course Code:	**BCS405B**	**CIE MARKS**	**50**
(L :T : P): SL	**(3:0:0) + (45 Hours / Sem)**	**SEE MARKS**	**50**
Credits:	**03**	**EXAM HOURS**	**03**

COURSE LEARNING OBJECTIVES:

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

1. Understand the basic concepts of graphs and their properties, and operations of graphs.
2. Hamiltonian and Euler graphs, trees and matrix representation of the graph.
3. Apply the concepts of a planar graph, matching and colouring in computer science engineering.

Module-1 **9 Hours**

Introduction. Basic definition, Application of graphs, finite, infinite and bipartite graphs, Incidence and Degree, Isolated vertex, pendant vertex and Null graph. Paths and circuits, Isomorphism, sub-graphs, walks, paths and circuits, connected graphs, disconnected graphs and components.

Text 1: Ch- 1.1 to 1.5, Ch - 2.1, 2.2, 2.4, 2.5

Module-2 **9 Hours**

Eulerian and Hamiltonian graphs, Euler graphs, Operations on graphs, Hamiltonian paths and circuits, Travelling salesman problem, directed graphs, types of digraphs, Digraphs and binary relation.

Text 1: Ch- 2.6 - 2.10

Module-3 **9 Hours**

Trees-properties, pendant vertex, Distance and centers in a tree, Rooted and binary trees, counting trees, spanning trees.

Connectivity Graphs-Vertex Connectivity, Edge Connectivity, Cut set and Cut Vertices, Fundamental circuits and cut-sets.

Text 1: Ch - 3.1 to 3.7, 4.1, 4.3 and 4.5.

Module-4 **9 Hours**

Planar Graphs-Planar graphs, Kuratowski's theorem (proof not required), Different representations of planar graphs, Euler's theorem, Geometric dual.

Graph Representations: Matrix representation of graphs, Adjacency matrix, Incidence Matrix, Circuit Matrix, Path Matrix.

Text 1: Ch - 5.2 to 5.6, 7.1 to 7.3, 7.8 and7.9.

Module-5 **9 Hours**

Graph Colouring-Colouring, Chromatic number, Chromatic polynomial, Matchings, Coverings, Four colour problem and Five colour problem, Greedy colouring algorithm.

Text 1: Ch - 8.1, 8.3 to 8.6.

COURSE OUTCOMES:

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

CO1	**Explain** basic definitions and types of graphs used in modeling.
CO2	**Apply** solutions using directed graphs and network flows.
CO3	**Analyze** shortest path algorithms and spanning tree techniques.
CO4	**Evaluate** Graphical representation of different types of Matrices and the representation of Planar Graphs.
CO5	**Design** graph colouring, matching, and planar graphs in applications.

ASSESSMENT DETAILS (Both CIE and SEE)

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. Narsingh Deo, Graph theory with the applications to engineering & Computer Science, Dovers Publications, 2016
2. J.A. Bondy and U.S.R. Murty. Graph theory with Applications, Springer, 2008.

Reference Books

1. Garry Chartand and Ping Zhang, Introduction to Graph Theory, Tata McGraw-Hill, 2006.
2. Frank Harary, Graph Theory, Narosa Publishing House, Latest edition.
3. R. Diestel, Graph Theory, free online edition, 2016: diestel-graph-theory.com/basic.html.

Web links and Video Lectures (e-Resources): VTU EDUSAT Program

1. http://nptel.ac.in/courses.php?disciplineID=111
2. http://www.class-central.com/subject/math(MOOCs)
3. http://academicearth.org/
4. VTU e-Shikshana Program

B.E. (CSE and Allied Branches)
Outcome Based Education(OBE) and Choice Based Credit System(CBCS)
SEMESTER – IV

Course Title:	**Metric Spaces**		
Course Code:	**BCS405C**	**CIE MARKS**	**50**
(L :T : P): SL	**(3:0:0) + (45 Hours / Sem)**	**SEE MARKS**	**50**
Credits:	**03**	**EXAM HOURS**	**03**

COURSE LEARNING OBJECTIVES:

1. Provide insight into the theory of sets
2. Learn basic concepts of metric spaces
3. Understand the concepts of connected sets and compact spaces

Module-1 **9 Hours**

Finite and infinite sets, countable and uncountable sets, cardinality of sets, Schroder-Bernstein theorem, cantor's theorem, Order relation in cardinal numbers, Arithmetic of cardinal numbers, Partially ordered set, Zorn's lemma and axioms of choice, various set-theoretic paradoxes.

Textbook 1 0.1–0.4 (Appendix/Preliminaries) **Reference 3**: 1.1, 1.6–1.8, 1.9

Module-2 **9 Hours**

Definition and examples of metric spaces, Open spheres and Closed spheres, Neighborhoods, Open sets, Interior, Exterior and boundary points, Closed sets, Limit points and isolated points, Interior and closure of a set, Boundary of a set, Bounded sets, Distance between two sets, Diameter of a set.

Textbook 1: 1.1, 2.1–2.3, 3.1–3.4
Textbook 2: 1.1, 2.1–2.5, 3.1–3.3

Module-3 **9 Hours**

Cauchy and Convergent sequences, Completeness of metric spaces, Cantor's intersection theorem, Dense sets and separable spaces, nowhere dense sets and Baire's category theorem, continuous and uniformly continuous functions, Homeomorphism. Banach contraction principle.
Textbook 1: 4.1–4.3, 5.1–5.4, 6.1, 7.1–7.3
Textbook 2: 4.1–4.4, 5.1–6.3, 9.1–9.2

Module-4 **9 Hours**

Compact spaces, Sequential compactness, Bolzano-Weierstrass property, Compactness and finite intersection property, Heine-Borel theorem, Totally bounded set, equivalence of compactness and sequential compactness.
Textbook 1: 8.1–8.5
Textbook 2: 7.1–7.4

Module-5 **9 Hours**

Separated sets, Disconnected and connected sets, components, connected subsets of R, Continuous functions on connected sets. Local connectedness and arc-wise connectedness.
Textbook 1: 9.1–9.3
Textbook 2: 8.1–8.4

COURSE OUTCOMES:

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

CO1	**Explain** the basic concepts of metric spaces and normed spaces.
CO2	**Apply** the concept of open, closed, and compact sets in metric spaces.
CO3	**Analyze** convergence, continuity, and completeness
CO4	**Evaluate** the role of completeness and contraction mappings.
CO5	**Apply** Baire's Theorem and Heine-Borel Theorem in analysis.

ASSESSMENT DETAILS (Both CIE and SEE)			
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. P.K. Jain & Khalil Ahamad, "*Metric Spaces"*. Narosa, 2019.
2. Micheal O; Searcoid, "Metric spaces". Springer-Verlag, 2009.

Reference Books

1. Satish Shirali & Harikishan L. Vasudeva, "*Metric Spaces"*, Springer-Verlag, 2006.
2. E.T. Copson, "*Metric spaces"*, Cambridge University Press, 1988.
3. P.R. Halmos, "*Naive Set Theor*y". Springer, 1974.
4. S. Kumaresan, "*Topology of Metric spaces",* 2nd edition, Narosa, 2011.

B.E. (CSE and Allied Branches)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – IV

Course Title:	**Game Theory**		
Course Code:	**BCS405D**	**CIE MARKS**	**50**
(L :T : P): SL	**(3:0:0) + (45 Hours / Sem)**	**SEE MARKS**	**50**
Credits:	**03**	**EXAM HOURS**	**03**

COURSE LEARNING OBJECTIVES:

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

1. Comprehend the basics of strategic gaming and mixed strategic equilibrium.
2. Enable students to develop skills on extensive gaming strategies.
3. Analyze and discuss various gaming models.
4. Illustrate some real-time situations.

Module-1 **9 Hours**

Introduction to Strategic Games: What is game theory? The theory of rational choice, Strategic games; Examples: The prisoner's dilemma, Bach or Stravinsky, Matching pennies; Nash equilibrium; Examples of Nash equilibrium; Best response functions; Dominated actions.

Textbook 1: Ch – 1.1 to 1.2, 2.1 to 2.4, 2.6 to 2.9

Module-2 **9 Hours**

Introduction: Strategic games in which players may randomize; Mixed strategy Nash equilibrium; Dominated actions; Pure equilibrium when randomization is allowed. Illustration: Expert Diagnosis; Equilibrium in a single population.

Textbook 1: Ch – 4.1 to 4.7

Module-3 **9 Hours**

Extensive games with perfect information; Strategies and outcomes; Nash equilibrium; Sub- game perfect equilibrium; Finding sub-game perfect equilibria of finite horizon games: Backward induction; Illustrations: The ultimatum game, Stackelberg's model of duopoly.

Textbook 1: Ch – 5.2 to 5.6, 6.2 to 6.3

Module-4 **9 Hours**

Bayesian Games**,** Motivational examples; General definitions; Two examples concerning information; Illustrations: Cournot's duopoly game with imperfect information, Providing a public good; Auctions: Auctions with an arbitrary distribution of valuations.

Textbook 1: Ch – 9.1 to 9.7, 9.9

Module-5 **9 Hours**

Competitive Games: Strictly competitive games and maximization.

Repeated games: The main idea; Preferences; Repeated games; Finitely and infinitely repeated Prisoner's dilemma; Strategies in an infinitely repeated Prisoner's dilemma; Nash equilibrium of an infinitely repeated Prisoner's dilemma, Nash equilibrium payoffs of an infinitely repeated Prisoner's dilemma.

Textbook 1: Ch – 11.3, 14.1 to 14.6

COURSE OUTCOMES

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

CO1	**Describe** basic game theory concepts including utility, payoff, and strategy.
CO2	**Apply** two-player zero-sum game models using different strategies.
CO3	**Analyze** Nash equilibrium and mixed strategy solutions.
CO4	**Evaluate** dominant strategies and minimax approaches in games
CO5	**Construct** models for real-world strategic situations using game theory.

ASSESSMENT DETAILS (Both CIE and SEE)

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. **Martin J. Osborne:** *"An Introduction to Game Theory"*, Oxford University Press, 1st Ed., 2003.
2. **Martin J. Osborne and Ariel Rubinstein**: *"A Course in Game Theory"*, MIT Press, 1st Ed., 1994

Reference Books

1. **John von Neumann and Oskar Morgenstern**: *"Theory of Games and Economic Behavior"*, Princeton University Press, 60th Anniversary Ed., 2007.
2. **R. Duncan Luce and Howard Raiffa**: *"Games and Decisions: Introduction and Critical Survey"*, Dover Publications, Reprint Ed., 1989.
3. **Roger B. Myerson**: *"Game Theory: Analysis of Conflict"*, Harvard University Press, 1st Ed., 1997.
4. **Robert Gibbons**: *"Game Theory for Applied Economists"*, Princeton University Press, 1st Ed., 1992.

Web links and Video Lectures (e-Resources):

1. http://nptel.ac.in/courses.php?disciplineID=111
2. http://www.class-central.com/subject/math(MOOCs)
3. http://academicearth.org/
4. VTU e- Shikshana Program
5. VTU EDUSAT Program.
6. http://www.themathpage.com/
7. http://www.abstractmath.org/
8. http://www.ocw.mit.edu/courses/mathematics/

Activity-Based Learning (Suggested Activities in Class)/Practical-Based Learning

- Quizzes
- Assignments
- Seminar

B.E. (Common to All branches) Outcome Based Education (OBE) and Choice Based Credit System (CBCS) SEMESTER – IV			
Course Title:	**Finance and Accounting**		
Course Code:	**BFAT456A**	**CIE MARKS**	**50**
(L: T: P) + S	**(1:0:0) + (15 HOURS / SEM)**	**SEE MARKS**	**50**
Credits:	**01**	**EXAM HOURS**	**02**

COURSE LEARNING OBJECTIVES:

1. To enable the students to understand the conceptual framework of accounting, reporting and financial statements.
2. To enable the students in understanding of books of accounts and accounting records leading to final accounts and interpretation there-off.
3. To acquaint the students with the concepts of Time value of Money for decision making.

Module-1	**3 Hours**

Introduction to Accounting: Meaning and objectives, Branches of Accounting, GAAP, IAS, IFRS, Concepts and Conventions, Users of accounting, accounting terminologies. (Only Theory)

Self-Learning Activity: Self-exploration exercise where students write about their understanding of Golden rules of Accounting, Classification of Assets and Liabilities.
Students can Compare GAAP, IAS, and IFRS side by side on Origin, purpose, scope, key differences.
Textbook 1: Section 1; Chapter 1, 2 and 3

Module-2	**3 Hours**

Journal, Ledgers, Trial balance, accounting equation, subsidiary books and final accounts (Only theory)

Self-Learning Activity: Students willwrite the liquidity and Profitability status by reading the financial statements of a Company.
Textbook 1: Section 1; Chapter 4, 5, 6 and 8

Module-3	**2 Hours**

Emerging accounting Practices: Window dressing, forensic accounting, Green Accounting, Human resource accounting and sustainability reporting.

Self-Learning Activity: Students will understand the significance of human resources accounting in the corporate environment.

Textbook 2: Chapter 6

Module-4	**3 Hours**

Overview of Financial Management: Meaning and scope- objectives of Financial Management-Functions of Finance-Role and functions of finance managers. Indian Financial System: Structure-types-Financial markets- Financial Instruments - Financial institutions and financial services- Non-Banking Financial Companies (NBFCs). (Theory only) **Self-Learning Activity**: Students will have to understand the classification of Banking industry and write the functions of Banking and Non -banking Organisations. **Textbook 3**: Part 1; Chapter 1 and 2
Module-5 **4 Hours**
Emerging areas in Financial Management: Risk Management- Behavioural Finance- Financial Engineering- Derivatives (Theory). **Time Value of Money:** Time value of money –Future value of single cash flow & annuity –Present value and discounting-present value of single cash flow, annuity & perpetuity. Simple interest & Compound interest & loan amortization schedule (Simple problems on FV, FV for Annuity, PV and PV for annuity & Amortization Schedule) **Self-Learning Activity**:Students will Write their own simple problems based on the concepts and solve them. Example: "Calculate the FV of $1,000 invested for 5 years at 6% compound interest." Students are engaged in preparing the Loan amortisation table with detailed interpretation. Students can Create mind maps showing the relationships and key components of risk management, financial engineering, derivatives, and behavioural finance. **Textbook 3**: Part III; Chapter 6

COURSE OUTCOMES:

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

CO1	Recall the basic accounting concepts and terminologies
CO2	Understanding of journal, ledger, trial balance subsidiary books and accounting Equation and final accounts.
CO3	Analyse emerging trends in accounting
CO4	Familiar with the Emerging areas of financial management and create a Loan Amortisation table.

ASSESSMENT DETAILS (Both CIE and SEE)

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Average of Two Internal Assessment Tests (MCQs) each conducted for 25 marks.
Alternate Assessment (B)	20	-	Any two Assessment methods as per 24RNBE4.2, scaled to 25 marks.
Total CIE (A+B)	50	20	The sum of two tests, two assignments, will be out of 100 marks and will be scaled down to 50 marks.
SEE (C)	50	18	Question papers having 50 MCQs with each question carrying 01 mark, and Examination duration is 2 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.			

Textbooks

1. Accounting forManagement, Maheswari S. N,Maheswari Sharad K, Vikas PublishingHouse (P) Ltd., Ed 2024.
2. Accounting for Managers, H.H. Ramesha, Pralhad Rathod, Praveen Kumar B. H., R.G. Saha, Himalaya Publishing House Pvt Ltd.,
3. Financial Management, Prasanna Chandra, TMH, Ed 2024

References

1. Financial Management, Ravi M Kishor, Taxmann's. (Recent Edition)
2. Financial Management, I M Paney, Vikas Publishing House Pvt. Ltd. (Recent Edition)

Accounting forManagement, Dr Jawahar Lal, Himalaya Publishing House Pvt Ltd., (Recent Edition)

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

Course Title:	**Introduction To Unix**		
Course Code:	**BCST456B**	**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 structure, architecture, and key features of the Unix/Linux operating system.
2. Gain hands-on experience with command-line interface (CLI) operations to efficiently navigate and manage Unix/Linux environments.
3. Develop and execute shell scripts to automate routine tasks and improve productivity.
4. Utilize Unix/Linux utilities for effective system administration, including process control, file manipulation, user management, and system monitoring.

Experiment-1

Write a shell script to execute essential Unix commands that perform the following tasks:

1. Display the current working directory.
2. List all files, including hidden files, in the current directory.
3. Sort the listed files by size in descending order.
4. Display the current date and time in full (detailed) format.
5. Display the name of the current logged-in user.

Experiment-2

Write a shell script to perform the following file and directory operations in Unix/Linux:

1. Create a nested directory structure, such as project/year2025/month07/week1.
2. Using a loop, create multiple text files (e.g., file1.txt, file2.txt, ..., file5.txt) inside the week1 directory.
3. Move and rename the files to a new directory named archive, renaming them as archived1.txt, archived2.txt, etc.
4. Copy the entire archive directory to a new location named backup.
5. Delete all .txt files from the archive directory using wildcard patterns.

Experiment-3

Write a shell script to demonstrate changing file permissions and ownership using both symbolic and numeric modes. The script should:

1. Create a sample file named example.txt.
2. Display the original permissions of the file using the long listing format (ls -l).
3. Change the file permissions using symbolic mode (e.g., add execute permission for the user).
4. Change the file permissions using numeric mode (e.g., set permissions to 754).
5. Display the updated permissions after each change using ls -l.

6. (Optional) Change the ownership of the file to another user (if the script is run with appropriate privileges).
7. Ensure that all permission changes are clearly shown before and after.

Experiment-4

Write a shell script to demonstrate the use of advanced text-processing commands in Unix/Linux. The script should perform the following operations on a structured text file named students.txt that contains tab-separated records in the format:

Roll_No	Name	Department	Marks
101	John	Computer	85
102	Ravi	Electronics	78
103	Asha	Computer	92
104	Maya	Electronics	85
105	Arjun	Mechanical	78

The script should perform and display the following tasks:

1. Display the contents of the file using cat.
2. Display the first 3 lines using head.
3. Display the last 2 lines using tail.
4. Extract and display only the names of the students using cut.
5. Convert all names to uppercase using tr.
6. Sort the data based on marks in ascending order using sort.
7. Display the list of unique departments using cut, sort, and uniq.
8. Use awk to print the names of students who scored more than 80 marks.
9. Use sed to replace the department name "Computer" with "CSE".

Ensure that each command's output is clearly labeled in the script.

Experiment-5

Write a shell script to demonstrate **input/output redirection** and the use of **pipelines** to process a text file and extract meaningful information. The script should utilize the following commands: grep, cut, tr, sort, uniq -c, and tee.

Perform the following tasks:

1. Use **input redirection** to read from a file named data.txt. The file contains comma-separated records in the format:
   pgsql
   CopyEdit
   ID,Name,Department
   101,John,Computer
   102,Ravi,Electronics
   103,Asha,Computer
   104,Maya,Electronics
   105,Arjun,Mechanical
   106,David,Computer
2. Use grep to extract only lines containing the word "Computer".
3. Use cut to extract the **Name** field from those lines.

4. Convert the extracted names to uppercase using tr.
5. Use tee to save the intermediate output to a file named filtered_names.txt.
6. Sort the names and use uniq -c to count the number of times each name appears.
7. Display the final output on the terminal.

Make sure to demonstrate both **input (<)** and **output (>, >>)** redirection, and use pipes (|) to chain commands.

Experiment-6

Write a shell script to perform advanced file searching and content inspection using Unix/Linux commands. The script should demonstrate the use of the find and grep commands with various options.

Perform the following tasks:

1. Recursively search from the current directory for:
   - Files with a specific **name** (e.g., report.txt).
   - Files of a specific **type** (e.g., directories or symbolic links).
   - Files with a specific **size** (e.g., greater than 1MB).
2. Search **recursively** for all files with the .log extension and use grep to:
   - Find and display lines containing the word "ERROR".
   - Display the filename along with the matching line.
3. Save the output of the error search to a file named error_report.txt.

Use appropriate options in find (like -name, -type, -size, etc.) and grep (like -r, -i, -n, --include=, etc.) to complete the task.

Ensure the script includes meaningful comments and clearly labeled output for each operation.

Experiment-7

Write a shell script to monitor and manage system processes effectively using Unix/Linux commands. The script should demonstrate the following tasks:

1. **Display a snapshot of current system processes** using commands like ps, top, or htop.
2. **List the top 5 memory-consuming processes**, showing details such as:
   - Process ID (PID)
   - User
   - % Memory usage
   - Command

   Use commands like ps aux with sorting and filtering.
3. **Start a long-running process (e.g., sleep or a dummy loop)** in the background and:
   - Display its background job ID and PID.
   - Monitor it using jobs and ps.
4. **Bring the background job to the foreground** using fg.
5. **Stop (suspend) and restart the process** using Ctrl+Z, bg, and kill.
6. **Cleanly terminate** the background process if it is still running.

The script logs the outputs clearly, and includes comments for each step to explain what is happening.

Experiment-8

Write a shell script to demonstrate **process monitoring** and **job control** in Unix/Linux. The script should perform the following tasks:

1. Display the current list of running processes using ps or top.
2. Identify and display the **top 5 memory-consuming processes**, including:
    - PID
    - USER
    - %MEM
    - COMMAND
3. Start a **dummy long-running process** (e.g., sleep 300) in the **background**.
4. Display the background job using the jobs command and retrieve its **PID** using ps.
5. Bring the background job to the **foreground** using fg.
6. Suspend the process using Ctrl+Z (or simulate with kill -STOP) and resume it in the background using bg.
7. Finally, **terminate** the process using kill.

The script should include **descriptive output** and **inline comments** to explain each step clearly.

Experiment-9

Write a shell script to demonstrate the use of **shell variables**, **environment variables**, the **export** command, and various **quoting mechanisms** in Unix/Linux. The script should perform the following tasks:

1. Create and display **shell variables** and perform simple arithmetic using them.
2. Use the **export** command to convert a shell variable into an **environment variable**, and verify it using printenv or env.
3. Demonstrate the difference between:
    - **Single quotes** ('...') – no variable or command substitution
    - **Double quotes** ("...") – allows variable and command substitution
    - **Backticks** (`...`) – used for command substitution
    - **$()** – preferred modern command substitution
4. Include examples that clearly show how the shell interprets quoted and unquoted text differently.

Experiment-10

Write a shell script that accepts a number as **user input** and performs the following checks:

1. Determine whether the number is **even** or **odd**.
2. Check if the number is **divisible by 3**.

Experiment-11

Write a shell script that performs the following tasks using **loops** and **file handling commands** in Unix/Linux:

1. Use a **loop** to print all **even numbers between 1 and 50**.
2. Traverse the current directory and **display the names of all files with a .txt extension**.

Experiment-12

Write a shell script to demonstrate the use of **disk**, **user**, and **group information** commands in Unix/Linux. The script should perform the following tasks using the specified commands:

1. Display all **mounted filesystems** along with their type and usage using df -Th.
2. Show the **disk usage** of the current user's **home directory** using du -sh ~.
3. List all users currently **logged in** using who, w, and users.
4. Display the current user's **UID, GID, and group memberships** using id and groups.

COURSE OUTCOMES:

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

CO1	Apply fundamental Unix/Linux commands for file, directory, and system operations.
CO2	Write shell scripts using variables, control structures, loops, and command substitution.
CO3	Use Unix utilities for process management, text processing, and file permissions.
CO4	Automate tasks and system monitoring using scheduling tools.

ASSESSMENT DETAILS (Both CIE and SEE)

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.			

Text Books:

1. Sumitabha Das , "Your UNIX/Linux: The Ultimate Guide" , 3rd edition, McGraw-Hill 2012

2. Sumitabha Das, "Unix: Concepts and Applications", 4th Edition, McGraw Hill, 2006.
3. W. Richard Stevens and Stephen A. Rago, "Advanced Programming in the UNIX Environment", 3rd Edition, Addison-Wesley, 2013.

Reference Books:

1. Behrouz A. Forouzan and Richard F. Gilberg, "Unix and Shell Programming", 1st Edition, Cengage Learning, 2003.
2. William E. Shotts Jr., "The Linux Command Line: A Complete Introduction", 2nd Edition, No Starch Press, 2019.
3. Free Software Foundation, "GNU Bash Reference Manual", Edition 5.2, GNU Project, 2022.

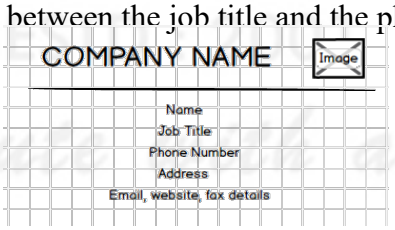
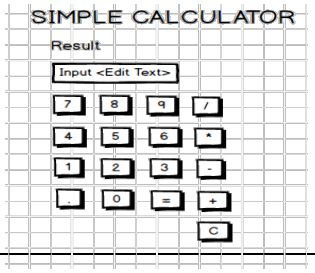
B.E. (Computer Science and Engineering)
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
SEMESTER – IV

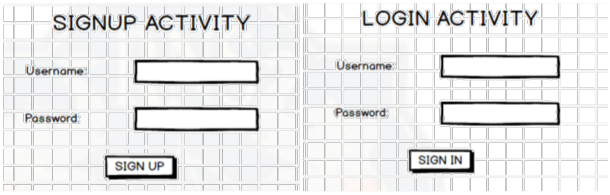
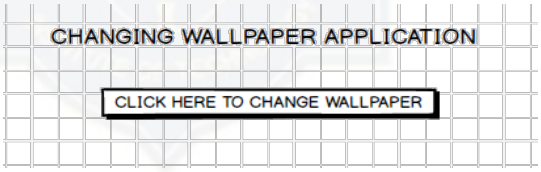
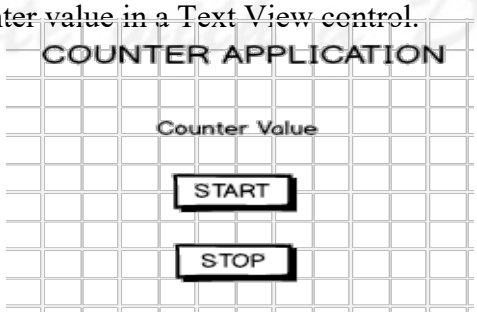
Course Title:	**Mobile App Development**		
Course Code:	**BCSL456C**	**CIE MARKS**	**50**
L : T : P	**0:0:2**	**SEE MARKS**	**50**
Credits:	**1**	**EXAM HOURS**	**3**

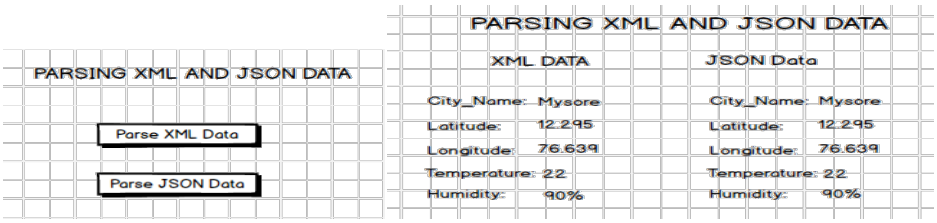
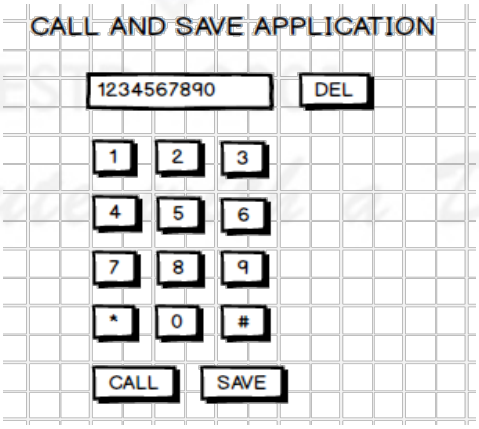
COURSE LEARNING OBJECTIVES:

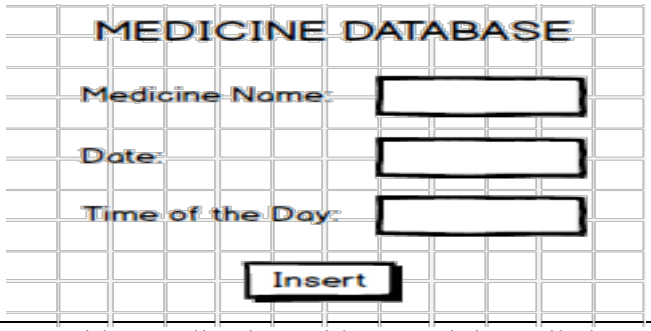
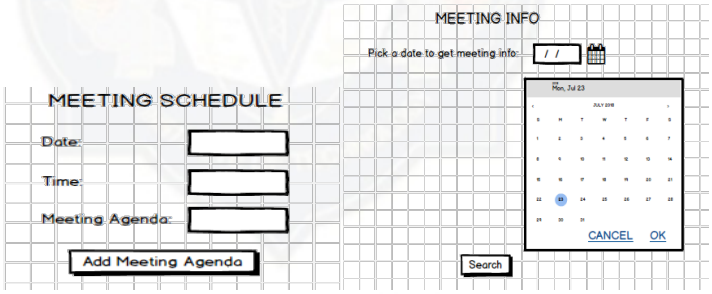
The purpose of the course is to facilitate the learner to:

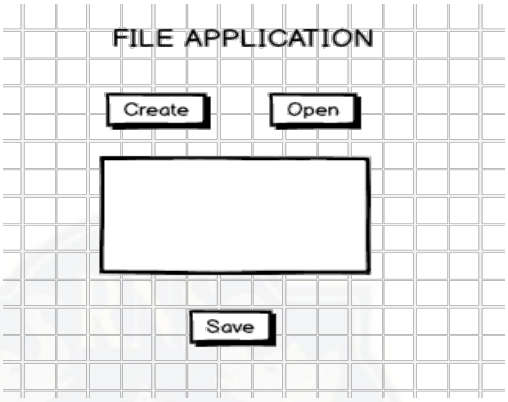
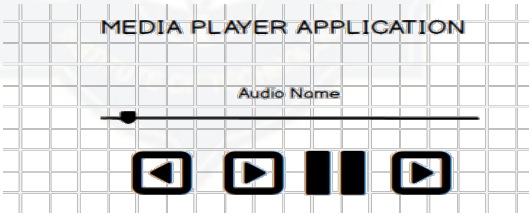
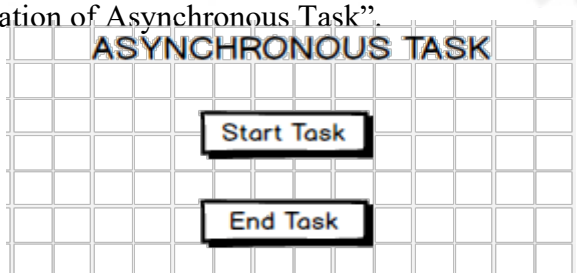
1. Develop basic Android applications by applying fundamental programming concepts and techniques.
2. Configure and use Android Studio IDE for designing, coding, and testing mobile applications.
3. Design and implement user interfaces using Android UI components such as layouts, widgets, and event handlers.
4. Create and manage local databases using SQLite for data storage and retrieval within Android apps.
5. Demonstrate data sharing techniques by implementing services, content providers, and inter-process communication in Android.

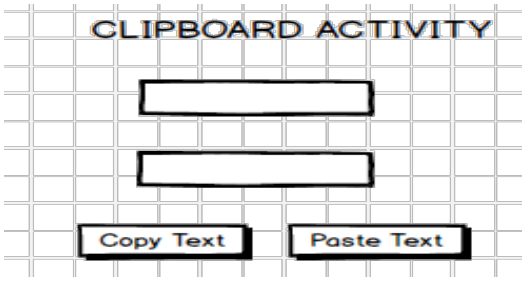
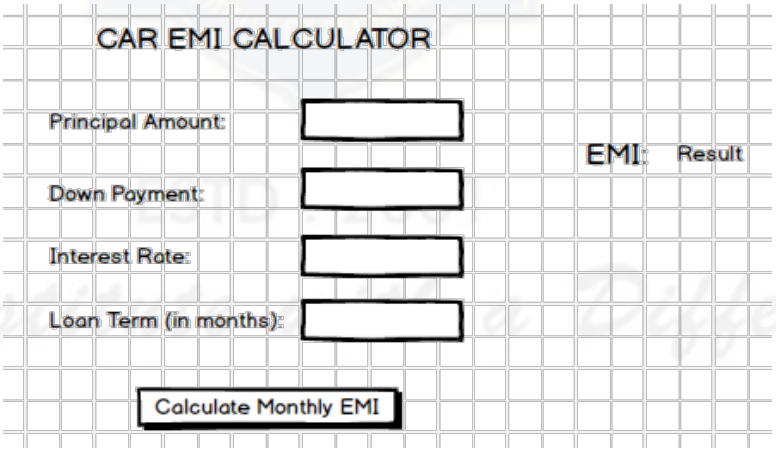
Sl. No.	EXPERIMENTS
	PART A
1	Create an application to design a Visiting Card. The Visiting card should have a company logo at the top right corner. The company name should be displayed in Capital letters, aligned to the Center. Information like the name of the employee, job title, phone number, address, email, fax and the website address is to be displayed. Insert a horizontal line between the job title and the phone number. 
2	Develop an Android application using controls like Button, Text View, Edit Text for designing a calculator having basic functionality like Addition, Subtraction, Multiplication and Division. 

3	Create a SIGN Up activity with Username and Password. Validation of password should happen based on the following rules: • Password should contain uppercase and lowercase letters. • Password should contain letters and numbers. • Password should contain special characters. • Minimum length of the password (the default value is 8). On successful SIGN UP proceed to the next Login activity. Here the user should SIGN IN using the Username and Password created during signup activity. If the Username and Password are matched then navigate to the next activity which displays a message saying "Successful Login" or else display a toast message saying "Login Failed". The user is given only two attempts and after that display a toast message saying "Failed Login Attempts" and disable the SIGN IN button. Use Bundle to transfer information from one activity to another. 
4	Develop an application to set an image as wallpaper. On click of a button, the wallpaper image should start to change randomly every 30 seconds. 
5	Write a program to create an activity with two buttons START and STOP. On pressing the START button, the activity must start the counter by displaying the numbers from One and the counter must keep on counting until the STOP button is pressed. Display the counter value in a Text View control. 

6	Create two files of XML and JSON type with values for City_Name, Latitude, Longitude, Temperature, and Humidity. Develop an application to create an activity with two buttons to parse the XML and JSON files which when clicked should display the data in their respective layouts side by side. 
7	Develop a simple application with one Edit Text so that the user can write some text in it. Create a button called “Convert Text to Speech” that converts the user input text into voice. 
8	Create an activity like a phone dialer with CALL and SAVE buttons. On pressing the CALL button, it must call the phone number and on pressing the SAVE button it must save the number to the phone contactsCreate an activity like a phone dialer with CALL and SAVE buttons. On pressing the CALL button, it must call the phone number and on pressing the SAVE button it must save the number to the phone contacts. 
PART B	

1	Write a program to enter Medicine Name, Date and Time of the Day as input from the user and store it in the SQLite database. Input for Time of the Day should be either Morning or Afternoon or Evening or Night. Trigger an alarm based on the Date and Time of the Day and display the Medicine Name. 
2	Develop a content provider application with an activity called "Meeting Schedule" which takes Date, Time and Meeting Agenda as input from the user and store this information into the SQLite database. Create another application with an activity called "Meeting Info" having Date Picker control, which on the selection of a date should display the Meeting Agenda information for that particular date, else it should display a toast message saying "No Meeting on this Date".
	
3	Create an application to receive an incoming SMS which is notified to the user. On clicking this SMS notification, the message content and the number should be displayed on the screen. Use appropriate emulator control to send the SMS message to your application. 

4	Write a program to create an activity having a Text box, and also Save, Open and Create buttons. The user has to write some text in the Text box. On pressing the Create button the text should be saved as a text file in MkSDcard. On subsequent changes to the text, the Save button should be pressed to store the latest content to the same file. On pressing the Open button, it should display the contents from the previously stored files in the Text box. If the user tries to save the contents in the Textbox to a file without creating it, then a toast message has to be displayed saying "First Create a File". 
5	Create an application to demonstrate a basic media player that allows the user to Forward, Backward, Play and Pause an audio. Also, make use of the indicator in the seek bar to move the audio forward or backward as required. 
6	Develop an application to demonstrate the use of Asynchronous tasks in android. The asynchronous task should implement the functionality of a simple moving banner. On pressing the Start Task button, the banner message should scroll from right to left. On pressing the Stop Task button, the banner message should stop. Let the banner message be "Demonstration of Asynchronous Task". 

7	Develop an application that makes use of the clipboard framework for copying and pasting of the text. The activity consists of two EditText controls and two Buttons to trigger the copy and paste functionality. 
8	Create an AIDL service that calculates Car Loan EMI. The formula to calculate EMI is E = P * (r(1+r)n)/((1+r)n-1) where E = The EMI payable on the car loan amount P = The Car loan Principal Amount r = The interest rate value computed on a monthly basis n = The loan tenure in the form of months The down payment amount has to be deducted from the principal amount paid towards buying the Car. Develop an application that makes use of this AIDL service to calculate the EMI. This application should have four EditText to read the Principal Amount, Down Payment, Interest Rate, Loan Term (in months) and a button named as "Calculate Monthly EMI". On click of this button, the result should be shown in a TextView. Also, calculate the EMI by varying the Loan Term and Interest Rate values. 

	Descriptions: 1. The installation procedure of the Android Studio/Java software must be demonstrated and carried out in groups. 2. Students should use the latest version of Android Studio/Java/ Kotlin to execute these programs. Diagrams given are for representational purposes only, students are expected to improvise on them. **3.** Part B programs should be developed as an application and are to be demonstrated as a mini project in a group by adding extra features or the students can also develop their application and demonstrate it as a mini-project. (Projects/programs are not limited to the list given in Part B).

COURSE OUTCOMES:

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

CO1	**Describe** the architecture and components of Android operating system.
CO2	**Build** simple mobile applications using Android Studio.
CO3	**Integrate** UI elements and layouts for user-friendly design.
CO4	**Develop** mobile apps using activities, intents, and fragments.
CO5	**Debug and test** Android applications on emulators and real devices.

ASSESSMENT DETAILS (Both CIE and SEE)

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. Google Developer Training Team, "Android Developer Fundamentals Course – Concept Reference", Google Developer Training, 1st Edition, 2017. https://www.gitbook.com/book/google-developer-training/android-developer-fundamentals- course-concepts/details.
2. Erik Hellman, "Android Programming – Pushing the Limits", Wiley India Pvt. Ltd., 1st Edition, 2014.
3. Pierre-Olivier Laurence, Amanda Hinchman-Dominguez, Mike Dunn, "Programming Android with Kotlin", O'Reilly SPD Publishers, 1st Edition, 2021.

Reference Books:

1. Bill Phillips, Chris Stewart, Kristin Marsicano, "Android Programming: The Big Nerd Ranch Guide", Big Nerd Ranch Guides, 3rd Edition, 2017.
2. Ian F. Darwin, "Android Cookbook: Problems and Solutions for Android Developers", O'Reilly Media, 2nd Edition, 2017.
3. Neil Smyth, "Android Studio 4.2 Development Essentials – Java Edition", Payload Media, 1st Edition, 2021.

B.E. (Computer Science and Engineering) Outcome Based Education (OBE) and Choice Based Credit System (CBCS) SEMESTER - IV			
Course Title:	**Advanced Java**		
Course Code:	**BCSL456D**	**CIE MARKS**	**50**
(L: T: P) + SL	**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. Understanding the fundamentals of collection framework
2. Demonstrate the fundamental concepts of String operations and Swing applications
3. Design and develop web applications using Java servlets and JSP
4. Apply database interaction through Java database Connectivity

Sl. No.	Experiments
1.	Implement a java program to demonstrate creating an ArrayList, adding elements, removing elements sorting elements of ArrayList.
2.	Develop a program to read random numbers between a range that are multiples of 2 sort the numbers according to tens place using comparator.
3.	Implement a java program to illustrate string user defined classes in collection.
4.	Implement a java program to illustrate different types of string class constructor.
5.	Implement a java program to illustrate the different types of character extraction and modification methods.
6.	Implement a java program to illustrate the use of different types of StringBuffer methods.
7.	Demonstrate a swing event handling application that creates 2 buttons say “Alpha” and “Beta” and display the text when button is clicked.
8.	Develop a program to display a greeting message on the browser, accept username from the client using servlet.
9.	Develop a servlet program to display name, usn, total marks by accepting student details.
10.	Develop a program to create and read the cookie from the given cookie name as “RNSIT” and its vale as “RNSIT123”.
11.	Develop a program to insert data into Student database and retrieve information base on particular queries (Update delete, search).
12.	Develop a program to design the login page and validating the user_id and password using JSP and database.

COURSE OUTCOMES:

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

CO1	Apply appropriate collection class/interface to solve the given problem
CO2	Demonstrate the concepts of String operations in Java
CO3	Apply the concepts of Swings to build Java applications
CO4	Develop web-based applications using Java servlets and JSP
CO5	Use JDBC to build database applications

ASSESSMENT DETAILS (Both CIE and SEE)

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Marks awarded for Weekly Conduction of Experiments and Submission of Laboratory records, scaled to 30 marks.
Alternate Assessment (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 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

4. Herbert Schildt: JAVA the Complete Reference. Twelfth Edition, Tata McGraw-Hill.
5. Jim Keogh, The Complete Reference J2EE, Tata McGraw-Hill 2007.
6. Kathy Sierra, Bert Bates, Headfirst Java, O'Reilly, 2nd edition.

Reference Books

1. Y. Daniel Liang: Introduction to JAVA Programming, 7th Edition, Pearson Education, 2007.
2. Stephanie Bodoff et al: The J2EE Tutorial, 2nd Edition, Pearson Education,2004.
3. Uttam K Roy, Advanced JAVA programming, Oxford University press, 2015.

B.E. (Common to All Branches) Outcome Based Education (OBE) and Choice Based Credit System (CBCS) SEMESTER – IV			
Course Title:	**ENVIRONMENTAL STUDIES**		
Course Code:	**BENS407 / 507**	**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. Introduce the basic concepts of ecology and ecosystem functioning, including biodiversity and biogeochemical cycles.
2. Understand various types of environmental pollution, their sources, impacts, and preventive measures.
3. Explore causes of environmental degradation and methods of conservation with special emphasis on Indian biodiversity.
4. Create awareness about climate change, global warming, environmental legislation, and significant environmental movements in India.

Module-1 **6 Hours**

Ecology: Basic concepts of ecosystem. Ecosystems (structure and function)- forest, desert, wetlands, river, oceanic and lake. Biotic and Abiotic components; Food chain; Ecological pyramid.

Biogeochemical Cycles: carbon cycle, nitrogen cycle, phosphorus cycle, water cycle

Sustainability: 17 Sustainable Development Goals (SDG)- history, targets, implementation, capacity development.

Module-2 **6 Hours**

Environmental Pollution: Surface and ground water pollution, noise pollution, soil pollution, air pollution and marine pollution - sources, impacts on living beings, impacts on environment, preventive measures. case-study on marine pollution.

Water Chemistry: Significance and effect of pH, total dissolved solids and hardness. Case study on impact of fluoride on human health.

Module-3 **6 Hours**

Environmental Degradation: Causes and effects of deforestation, desertification, soil erosion and glacier melting.

Environmental Conservation: Significance and key features of national parks, wildlife sanctuaries, reserved forest and mangrove forest. Importance of western ghats as biodiversity hotspot (flora and fauna)

Module-4 **6 Hours**

Indian Monsoon and Global Warming

Climate- Impact on monsoons, significance of El Nino and La Nina on India climate.

Global warming: causes, greenhouse gases, effects on environment and human health. Causes and effects of acid rain and ozone depletion. Air sampler and its importance in assessing pollutants.
Module-5 **6 Hours**
Environmental Legislation: Environmental Protection Act 1984: Definition, Significance and key features. **Indian Environmental Movements:** Bishnoi Movement, Chipko & Appiko movement.

COURSE OUTCOMES:

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

CO1	Describe the components and functions of ecosystems and explain major biogeochemical cycles and sustainability goals.
CO2	Understand various forms of pollution and environmental degradation, along with their impacts on ecosystems and human health.
CO3	Demonstrate awareness of climate issues, Indian environmental legislation, and the role of community-based movements in environmental protection.

ASSESSMENT DETAILS (Both CIE and SEE)

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	12	Average of Three Internal Assessment Tests each of 50 marks, scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	8	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 should 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 out of allotted 50 marks to appear for SEE, with minimum 40 % scored in every component of CIE.			

Textbooks

1. Benny Joseph, "Environmental studies", 3rd ed., Tata McGraw-Hill, 2017.
2. S M Prakash, "Environmental studies", 3rd ed., Pristine publishing house, 2018.

Reference Books: -

1. M.Anji Reddy, "Textbook of environmental science and Technology", revised ed., BS publications, 2014.
2. Dr. B.S Chauhan, "Environmental studies", 1st ed., University of science press, 2008.

B.E. (Common to all branches) Outcome Based Education (OBE) and Choice Based Credit System (CBCS) SEMESTER – IV / V			
Course Title:	**BIOLOGY FOR ENGINEERS**		
Course Code:	**BBOE407/507**	**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 the students with the basic biological concepts and their engineering applications.
2. Enable the students with an understanding of bio design principles to create novel devices and structures.
3. Provide the students an appreciation of how biological systems can be re-designed as substitute products for natural systems.
4. Motivate the students to develop 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**
Introduction to 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, food processing, enzymes in biobleaching, Lipids in biodiesel and detergents production.	
Module-3	**6 Hours**
Introduction to adaptations of anatomical principles for bioengineering design, Brain as a CPU system. Eye as a Camera system. Heart as a pump system. Lungs as purification system. Kidney as a filtration system.	
Module-4	**6 Hours**
Introduction to 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 perflourocarbons (PFCs).	
Module-5	**6 Hours**
Introduction to trends in bioengineering, Tissue engineering, stem cells and their applications, Bioprinting 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 the basic biological concepts for relevant industrial applications. adaptation.
CO3	Evaluate the principles of design and development, for exploring engineering knowledge in biological
CO4	Think critically towards exploring innovative biobased solutions for socially relevant problems.

ASSESSMENT DETAILS (Both CIE and SEE)

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	12	Average of Three Internal Assessment Tests each of 50 marks, scaled down to 30 marks.
Continuous and Comprehensive Evaluation (B)	20	8	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 should 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 out of allotted 50 marks to appear for SEE, with minimum 40 % scored in every component of CIE.			

Textbooks

1. Basuchandra's Biology for Engineers, Chandra shekara. B.M, Basavaraju B.C, 2025.
2. Biology for Engineers, Rajendra Singh C and Rathnakar Rao N, Rajendra Singh C and Rathnakar Rao N Publishing, Bengaluru, 2023.

Reference Books

1. Human Physiology, Stuart Fox, Krista Rompolski, McGraw-Hill eBook. 16th Edition, 2022
2. Biology for Engineers, Thyagarajan S., Selvamurugan N., Rajesh M.P., Nazeer R.A., Thilagaraj W.
3. Barathi S., and Jaganthan M.K., Tata McGraw-Hill, New Delhi, 2012.
4. Biology for Engineers, Arthur T. Johnson, CRC Press, Taylor and Francis, 2011
5. Biomedical Instrumentation, Leslie Cromwell, Prentice Hall 2011.

6. Biology for Engineers, Sohini Singh and Tanu Allen, Vayu Education of India, New Delhi, 2014.
7. Biomimetics: Nature-Based Innovation, Yoseph Bar-Cohen, 1st edition, 2012, CRC Press.
8. Bio-Inspired Artificial Intelligence: Theories, Methods and Technologies, D. Floreano and C. Mattiussi, MIT Press, 2008.
9. Bioremediation of heavy metals: bacterial participation, by C R Sunilkumar, N Geetha, A C Udayashankar, Lambert Academic Publishing, 2019.

B.E. (Common to All branches) Outcome Based Education (OBE) and Choice Based Credit System (CBCS) SEMESTER – IV			
Course Title:	**Universal Human Values (UHV)**		
Course Code:	**BUHV408**	**CIE MARKS**	**50**
(L: T: P) + S	**(1:0:0) + (15 HOURS / SEM)**	**SEE MARKS**	**50**
Credits:	**01**	**EXAM HOURS**	**02**

COURSE LEARNING OBJECTIVES:

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

4. To help the students appreciate the essential complementarity between 'VALUES' and 'SKILLS' to ensure sustained happiness and prosperity which are the core aspirations of all human beings.
5. To facilitate the development of a Holistic perspective among students towards life and profession as well as towards happiness and prosperity based on a correct understanding of the Human reality and the rest of existence. Such a holistic perspective forms the basis of Universal Human Values and movement towards value-based living in a natural way.
6. To highlight plausible implications of such a Holistic understanding in terms of ethical human conduct, trustful and mutually fulfilling human behaviour and mutually enriching interaction with Nature.
7. This course is intended to provide a much-needed orientation input in value education to the young enquiring minds.

Module-1	**3 Hours**

Introduction to Value Education: Right Understanding, Relationship and Physical Facility (Holistic Development and the Role of Education) Understanding Value Education, Self-exploration as the Process for Value Education, Continuous Happiness and Prosperity – the Basic Human Aspirations, Happiness and Prosperity – Current Scenario, Method to Fulfil the Basic Human Aspirations.

Self-Learning Activity: Self-exploration exercise where students write about their understanding of happiness and prosperity.

Module-2	**3 Hours**

Harmony in the Human Being: Understanding Human being as the Co-existence of the Self and the Body, distinguishing between the Needs of the Self and the Body, The Body as an Instrument of the Self, Understanding Harmony in the Self, Harmony of the Self with the Body, Programme to ensure self-regulation and Health.

Self-Learning Activity: Students will categorize their daily needs into 'self' and 'body' to understand their coexistence.

Module-3	**3 Hours**

Harmony in the Family and Society: Harmony in the Family – the Basic Unit of Human Interaction, 'Trust' – the Foundational Value in Relationship, 'Respect' – as the Right Evaluation, Other Feelings, Justice in Human-to- Human Relationship, Understanding Harmony in the Society, Vision for the Universal Human Order.

Self-Learning Activity: Students will write and share a personal story where trust or respect played a key role in a relationship.
Module-4 **3 Hours**
Harmony in the Nature/Existence: Understanding Harmony in the Nature, Interconnectedness, self-regulation and Mutual Fulfilment among the Four Orders of Nature, Realizing Existence as Co-existence at All Levels, The Holistic Perception of Harmony in Existence. **Self-Learning Activity**: Students will have a nature walk to observe and reflect on coexistence and mutual fulfilment in nature.
Module-5 **3 Hours**
Implications of the Holistic Understanding – a Look at Professional Ethics: Natural Acceptance of Human Values, Definitiveness of (Ethical) Human Conduct, A Basis for Humanistic Education, Humanistic Constitution and Universal Human Order, Competence in Professional Ethics Holistic Technologies, Production Systems and Management Models-Typical Case Studies, Strategies for Transition towards Value-based Life and Profession. **Self-Learning Activity**: Students are engaged in drafting their personal code of professional ethics aligned with human values.

COURSE OUTCOMES:

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

CO1	Identify and describe the role of self, family, society, and nature in achieving holistic human well-being.
CO2	Analyze real-life personal and societal problems through the lens of human values and ethical reasoning.
CO3	Apply human values to interpersonal and professional situations to ensure harmony in relationships and society.
CO4	Integrate sustainability, ethics, and lifelong learning into personal and professional practices.

ASSESSMENT DETAILS (Both CIE and SEE)

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
CIE – IA Tests (A)	30	-	Average of Two Internal Assessment Tests (MCQs) each conducted for 25 marks.
Alternate Assessment (B)	20	-	Any two Assessment methods as per 24RNBE4.2, scaled to 25 marks.
Total CIE (A+B)	50	20	The sum of two tests, two assignments, will be out of 100 marks and will be scaled down to 50 marks.

SEE (C)	50	18	Question papers having 50 MCQs with each question carrying 01 mark, and Examination duration is 2 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.			

Textbooks

1. The Textbook A Foundation Course in Human Values and Professional Ethics, R R Gaur, R Asthana, G P Bagaria, 2nd Revised Edition, Excel Books, New Delhi, 2019. ISBN 978-93-87034- 47-1
2. The Teacher's Manual for A Foundation Course in Human Values and Professional Ethics, R R Gaur, R Asthana, G

Reference Books

1. Jeevan Vidya: Ek Parichaya, A Nagaraj, Jeevan Vidya Prakashan, Amar kantak, 1999.
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi
5. Small is Beautiful - E. F Schumacher.
6. Slow is Beautiful - Cecile Andrews
7. Economy of Permanence - J C Kumarappa
8. Bharat Mein Angreji Raj – Pandit Sunderlal
9. Rediscovering India - by Dharampal
10. Hind Swaraj or Indian Home Rule - by Mohandas K. Gandhi
11. India Wins Freedom - Maulana Abdul Kalam Azad
12. Vivekananda - Romain Rolland (English)
13. Gandhi - Romain Rolland (English)
14. Sussan George, 1976, How the Other Half Dies, Penguin Press. Reprinted 1986, 1991
15. Donella H. Meadows, Dennis L. Meadows, Jorgen Randers, William W. Behrens III, 1972, Limits to Growth – Club of Rome's report, Universe Books.
16. A Nagraj, 1998, Jeevan Vidya Ek Parichay, Divya Path Sansthan, Amarkantak.
17. P L Dhar, RR Gaur, 1990, Science and Humanism, Commonwealth Publishers.
18. A N Tripathy, 2003, Human Values, New Age International Publishers.
19. Subhas Palekar, 2000, How to practice Natural Farming, Pracheen (Vaidik) Krishi Tantra Shodh, Amravati.

20. E G Seebauer & Robert L. Berry, 2000, Fundamentals of Ethics for Scientists & Engineers, Oxford University Press
21. M Govindrajan, S Natrajan & V.S. Senthil Kumar, Engineering Ethics (including Human Values), Eastern Economy Edition, Prentice Hall of India Ltd.
22. B P Banerjee, 2005, Foundations of Ethics and Management, Excel Books.
23. B L Bajpai, 2004, Indian Ethos and Modern Management, New Royal Book Co., Lucknow. Reprinted 2008.

B.E. (Common to All Branches) Outcome Based Education (OBE) and Choice Based Credit System (CBCS) SEMESTER – IV			
Course Title:	**NSS-National Service Scheme**		
Course Code:	**BNSK459**	**CIE MARKS**	**100**
L : T : P	**0:0:0:30**	**SEE MARKS**	**0**
Credits:	**0**	**EXAM HOURS**	**-**

COURSE LEARNING OBJECTIVES:

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

1. Understand the community in general in which they work.
2. Identify the needs and problems of the community and involve them in problem – solving.
3. Develop among themselves a sense of social & civic responsibility & utilize their knowledge in finding practical solutions to individual and community problems.
4. Develop competence required for group-living and sharing of responsibilities & gain skills in mobilizing community participation to acquire leadership qualities and democratic attitudes.
5. Develop capacity to meet emergencies and natural disasters & practice national integration and social harmony in general.

Module-1
Water conservation techniques – Role of different stakeholders– Implementation.
Module-2
Preparing an actionable business proposal for enhancing the village income and approach for implementation.
Module-3
Helping local schools to achieve good results and enhance their enrolment in Higher/technical/ vocational education.
Module-4
NSS Activity-1
Module-5
NSS Activity-2

COURSE OUTCOMES:

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

CO1	Understand the significance of individual responsibilities towards society and contribute meaningfully to community welfare.
CO2	Analyze environmental and societal issues, and design feasible solutions to address them effectively
CO3	Critically evaluate existing systems and propose practical, sustainable solutions for long-term development.
CO4	Plan and implement government initiatives or self-driven projects effectively in real-life community settings.
CO5	Develop the capacity to respond to emergencies and natural disasters, while practicing national integration, social harmony, and unity in diversity.

ASSESSMENT DETAILS (Both CIE and SEE)

Assessment Type	Maximum Marks	Minimum Passing Marks	Evaluation Details
Evaluation based on Reports (A)	100	-	5 modules Reports, 5 modules X 10 M each = 50 M
Total CIE Theory (A)	**100**	**40**	
SEE (B)	**-**	**-**	**-**
CIE+SEE (A+B)	**100**	**40**	**Final marks shall be the sum of marks scored in Components A, B**
• The minimum marks to be secured in CIE is 40 out of allotted 50 marks with minimum 40 % scored in every component of CIE. • No SEE for this subject			

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.

Weblinks:

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

B.E. (Common to All Branches) Additional Mathematics-II (Mandatory Learning Course) A bridge course for Lateral Entry Students			
Course Code:	**BMATDIP41**	CIE Marks	100
(L:T:P) + SL	(3:0:0) + (45 Hours/Sem)	Total Marks	100
Credits	0	Exam Hours	3

Course objectives:

The mandatory course **BMATDIP41** viz., **Additional Mathematics –II** aims to provide essential concepts of Linear algebra, Elementary Statistics, insight into Elementary probability theory and Numerical methods.

Module-1	**8 Hours**
Introduction, Rank of a matrix by elementary row operations, Consistency of system of linear equations, Solution by Gauss Elimination method. Eigenvalues and eigenvectors of a square matrix. Problems.	
Module-2	**8 Hours**
Curve Fitting: Curve fitting by the method of least squares- fitting of the curves of the form, $y = ax + b$, $y = ax^2 + bx + c$ and $y = ae^{bx}$. Correlation-Karl Pearson's coefficient of correlation-problems. Regression analysis, lines of regression (without proof) –Problems	
Module-3	**8 Hours**
Introduction, Sample space and Events, Axioms of Probability. Addition and Multiplication theorem. Conditional Probability. Independent events. Baye's theorem, Problems.	
Module-4	**8 Hours**
Finite differences, Interpolation/extrapolation using Newton's forward and Backward difference formulae (No derivation), Problems. Solution of polynomial and transcendental equations by Newton–Raphson and Regula–Falsi methods (no derivation), Problems. Numerical Integration: Simson's 1/3 rd rule and 3/8 rule, problems.	
Module-5	**8 Hours**
Numerical solution of first-order ordinary differential equations: Taylor's series method, Modified Euler's method, Runge-Kutta method of order 4, Milne's predictor-corrector method. Problems.	

Course outcome:

At the end of the course the student will be able to:

CO1	Test for consistency and solving the system of linear equations
CO2	Make use of correlation and regression analysis to fit a suitable mathematical model for statistical data.
CO3	Apply elementary probability theory and solve related problems

CO4	To interpolate/extrapolate from the given data
CO5	Apply knowledge of numerical methods in modelling and solving engineering problems

Assessment Details (CIE) Continuous Internal Evaluation:

- Three Unit Tests each of **20 Marks (duration 01 hour**)
    1. The first test at the end of 5th week of the semester
    2. The second test at the end of the 10th week of the semester
    3. Third test at the end of the 15th week of the semester
    4. Two assignments each of **20 Marks**
    5. First assignment at the end of the 4th week of the semester
    6. Second assignment at the end of the 9th week of the semester
- The sum of three tests and two assignments will be out of 100 marks.
- The student shall secure a minimum of 40% of marks of the course to qualify and become eligible for the award of a degree.

Textbook

1. Higher Engineering Mathematics: B. S. Grewal, Khanna Publishers, New Delhi, 43rd Ed., 2015.

Reference Books:

1. Higher Engineering Mathematics: V. Ramana, McGraw-Hill Education, 11th Ed.
2. Engineering Mathematics: Srimanta Pal & Subodh C. Bhunia, Oxford University Press, 3rd Reprint, 2016.
3. A textbook of Engineering Mathematics: N.P Bali and Manish Goyal, Laxmi Publications, Latest edition.
4. Higher Engineering Mathematics: H.K. Dass and Er. Rajnish Verma, S. Chand Publication (2014).

Web links and Video Lectures (e-Resources):

1. http://www.class-central.com/subject/math (MOOCs)
2. http://academicearth.org/
3. http://www.bookstreet.in.
4. VTU e-Shikshana Program
5. VTU EDUSAT Program

Activity-Based Learning (Suggested Activities in Class)/ Practical Based learning

- Quizzes
- Assignments
- Seminars

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:
Channasandra, Dr. Vishnuvardhan Road, Bengaluru - 560 098
Ph: (080) 28611880, 28611881