Royal College of Arts, Science and Commerce (Autonomous)

Affiliated to University of Mumbai

Program: Bachelor of Science

Course: Computer Science

Syllabus for Semester: I and II

Amendments of FYBSC (SEM-I and SEM-II) Computer Science Practical course

Syllabus for Undergraduate Programme as per

National Education Policy (NEP-2020) with effect from the academic year 2025-2026

Principal
ROYAL COLLEGE OF ARTS
SCIENCE & COMMERCE
PENKAR PADA, MIRA ROAD,
DIST : THANE. PIN : 401107.

Scanned with Cam

NEP Credit Structure for Computer Science Department

NEP Credit Structure for Computer Science Department

Level	Sem	Major		Minor	OE	VSC	SEC	AEC	IKS	VEC	OJT/FP /RP/CC /CEP	Cumulative Credits	
		DSC	DSE										
4.5	I	4TH+2PR 6		2TH	2TH+ 2TH	2PR	2PR	2TH	2TH	2TH		22	**UG Certificate Cumulative Credit:44**
	II	4TH+2PR 6		2TH	2TH+ 2TH	2PR	2PR	2TH		2TH	2 CC	22	
Exit Option: Award of UG Certificate in Major with 40-44 Credits and an Additional 4 Credits Core NSQF Course / Internship OR Continue with Major and Minor													
5	III	8TH+2PR 10		2TH+2PR 4	2TH		2PR	2TH			2 CC	22	**UG Diploma Cumulative Credit:88**
	IV	8TH+2PR 10		2TH+2PR 4	2TH		2PR	2TH			2 CEP	22	
Exit Option: Award of' UG Diploma in Major and Minor with 80-88 Credits and an Additional 4 Credits Core NSQF Course /Internship OR Continue with Major and Minor													
5.5	V	8TH+2PR 10	2	2TH+2TH 4		2					2 FP +2 OJT	22	**UG Degree Cumulative Credit:132**
	VI	8TH+2PR 10	2	2TH+2TH 4		2					2 FP +2 OJT	22	
Total		**52**	**4**	**20**	**12**	**8**	**8**	**8**	**2**	**4**	**14**	**132**	

Semester – I				
Course Code	**Course Type**	**Course Title**	**Credits**	**Lectures /Week**
RUCSMJ101	Major I	Digital Systems	2	2
RUCSMJ102	Major II	Database Management System	2	2
RUCSMJP1	**Major Practical**	**Computer Science Practical 1**	**2**	**4**
RUCSMN103	Minor	Discrete Mathematics	2	2
RUCSOE104	OE	Open Elective OE I	2	2
RUCSOE105	OE	Open Elective OE II	2	2
RUCSVSC106	VSC	Linux Programming	2	2
RUCSSEC107	SEC	Python Programming	2	2
RUCSAEC108	AEC	Communication Skills	2	2
RUCSVEC109	VEC	Human Values	2	2
RUCSIKS110	IKS	Development in Science and Technology	2	2

[Abbreviation - OE – Open Electives, VSC – Vocation Skill Course, SEC – Skill Enhancement Course, (VSEC), AEC – Ability Enhancement Course, VEC – Value Education Course, IKS – Indian Knowledge System, OJT – on Job Training, FP – Field Project, CEP – Continuing Education Program, CC – Co-Curricular, RP – Research Project]

Semester – II				
Course Code	**Course Type**	**Course Title**	**Credits**	**Lectures / Week**
RUCSMJ201	Major I	Object Oriented Programming	2	2
RUCSMJ202	Major II	Design and Analysis of Algorithm	2	2
RUCSMJP2	**Major Practical**	**Computer Science Practical 2**	**2**	**4**
RUCSMN203	Minor	Descriptive Statistics	2	2
RUCSOE204	OE	Open Elective OE I	2	2
RUCSOE205	OE	Open Elective OE II	2	2
RUCSVSC206	VSC	Web Technologies	2	2
RUCSSEC207	SEC	Application Development using Python	2	2
RUCSAEC208	AEC	Business Communication	2	2
RUCSVEC209	VEC	Green Technologies	2	2
RUCSCC210	CC	NSS, NCC, Sports, Cultural, Yoga, Music, Performing Arts, DLLE	2	2

Programme Outcomes (POs) for Computer Science

	On completing B.Sc. Computer Science, the student will be able to:
PO1	Develop understanding and knowledge of the fundamental theories, applications, technologies in computer science.
PO2	Understanding and applying the knowledge of networking, web design, security, cloud computing, IoT.
PO3	Understanding of best practices, standards to develop user interactive and abstract applications.
PO4	Demonstrate proficiency in using current techniques, skills, and tools necessary for computing practice.
PO5	Apply concepts, principles and theories relating to computer science to new situations.
PO6	Apply standard software engineering practices and strategies in real-time software project development.
PO7	Apply technical skills to formulate & design solutions to solve real world problems.
PO8	Pursue higher studies of specialization and to take up technical employment.
PO9	Develop Work Ethics , Communication Skills to present their work effectively and coherently.
PO10	Engage in independent and life-long learning in the background of the rapidly changing IT industry.

Programme Specific Outcomes (PSOs) for Computer Science

	On completing B.Sc. Computer Science, the student will be able to:
PSO1	Demonstrate proficiency in different computing paradigms needed for a proper understanding of computer science.
PSO2	Develop and implement software solutions to effectively address real-world problems in business, science, and social contexts.
PSO3	Build up programming, analytical and logical thinking abilities.
PSO4	Design, develop and analyze the use of the Information security tools and technique for providing security.
PSO5	Demonstrate programs in various languages to enhance computer and internet capabilities.
PSO6	Apply the standard software engineering practices and strategies in real time software project development.

Course/ Paper Title	**Digital Systems**
Course offered as	**Major I**
Course Code	**RUCSMJ101**
Semester	**I**
No. of Credits	**2 (30 hours)**
No. of lecture Hours/week	**2**

Sr No.	Course Learning Objectives:
CLO1	Describe the basic structure and functions of a computer system and logic gates.
CLO2	Summarize the working principles of digital gates, flip-flops, adders, multiplexers and counters.
CLO3	Classify memory systems, including cache, virtual memory, and RAID levels
CLO4	Apply Boolean algebra to simplify logic functions using KMAP, number system conversion.
CLO5	Describe control unit function and design a basic control unit using both hardwired and micro-programmed techniques.
CLO6	Explain processor organization and advanced computer architecture, including multiprocessor systems and multi-core designs.
CLO7	Classify array processors according to Flynn's taxonomy.

Course Outcome

	On completing the course, the student will be able to:
CO1	Explain the basic structure and operation of computers and their functional units and logic gates.
CO2	Perform conversions between different number systems and apply Boolean algebra to simplify logical expressions.
CO3	Design combinational and sequential circuits for various digital logic applications.
CO4	Compare and contrast different computer architecture.
CO5	Classify and describe the different types of memory systems, cache memory performance, and array processors.
CO6	Explain and differentiate between different RAID levels, types of processors, and parallel processing systems.

Detailed Syllabus

Module	Title with content	No. of lectures
I	**Fundamentals of Computer & Digital Logic:** Basic structure and operation of a computer, functional units and their interaction. Representation of numbers systems and Conversions, Boolean algebra. **Logic & Universal Gates**: Basic gates: NOT, AND, OR. Universal Gates: NAND & NOR Gates, Simplification of Logic Circuits using Karnaugh Maps, Tristate buffers. **Combinational and Sequential Circuits:** Adders: half Adder & Full Adder, Mux, De-Mux, Comparator. Flip Flops (SR, JK & D), Counters: synchronous and asynchronous Counter, Shift registers. **Computer Architecture & Organization:** Comparison of Computer Organization & Architecture, Interconnection Structures. Bus Interconnections, Input / Output: I/O Module, Programmed I/O, Interrupt Driven I/O, Direct Memory Access, Instruction Formats, Instruction Sets, Addressing Modes, Addressing Modes Examples with Assembly Language [8085/8086 CPU], Basic Microprocessor operations: Data Transfer (Register / Memory) Operations, Arithmetic & Logical Operations, Instruction Cycle, Instruction Pipelining. Introduction to RISC and CISC Architecture.	**15**
II	**Memory System Organization:** Classification of main memory and design parameters, Memory Hierarchy, Internal Memory: RAM, ROM, SRAM and DRAM, Cache Memory: Design Principles, Memory mappings, Replacement Algorithms: FIFO, LRU, LFU. Cache performance. Virtual Memory, External Memory: Magnetic Discs, Optical Memory, Flash Memories. **RAID Levels**: RAID level 0 – 6 and their Comparisons. **Processor Organization:** Structure and Function. Types of Registers. Control unit, Hardwired Implementation, Micro-programmed Control. **Fundamentals of Advanced Computer Architecture:** Classification of Parallel Systems, Flynn‟s Taxonomy, Array Processors, Clusters, and NUMA Computers. Multiprocessor Systems: Structure & Interconnection Networks, Symmetric and Asymmetric multiprocessing, Multi-Core Computers: Introduction, Organization and Performance.	**15**

References:

1. M. Mano, Computer System Architecture 3rd edition, Pearson.
2. Carl Hamacher et al., Computer Organization and Embedded Systems, 6 ed., McGraw-Hill 2012
3. R P Jain, Modern Digital Electronics, Tata McGraw Hill Education Pvt. Ltd. , 4th Edition, 2010

Additional References:

1. William Stallings (2010), Computer Organization and Architecture-designing for performance,8th edition, Prentice Hall, New Jersey.
2. Anrew S. Tanenbaum (2006), Structured Computer Organization,5th edition,Pearson Education Inc,
3. John P. Hayes (1998), Computer Architecture and Organization, 3rd edition, Tata McGrawHill
4. Ramesh Gaonkar (2013), Microprocessor Architecture, Programming and Application with 8085, 6th edition, Penram.

Evaluation for Theory Courses (2 Credit Courses)

I. Continuous Internal Evaluation (CIA) - 20 Marks (40%)	**II. Semester End Theory Examination (SEE)- 30 Marks (60%)**
(i) Mid-Term Class Test – 10 Marks **(ii) Assignment/ Case study/ Presentations – 10 Marks**	**A Semester End Theory Examination of 1 hour duration for 30 Marks as per the paper pattern given below.**

Format of Semester End Theory question paper pattern

Question	**Based on**	**Options**	**Marks**
Q.1	Module I	Any 2 out of 4	10
Q.2	Module II	Any 2 out of 4	10
Q.3	Module I & II	Any 2 out of 4	10
		Total	30

- All questions shall be compulsory with internal choice within the questions.
- Each Question may be subdivided into sub questions as a, b, c, d, etc. & each sub question carries 5 Marks.

Course/ Paper Title	**Database Management System**
Course offered as	**Major II**
Course Code	**RUCSMJ102**
Semester	**I**
No. of Credits	**2 (30 hours)**
No. of lecture Hours/week	**2**

Sr No.	Course Learning Objectives:
CLO1	To make students aware of the fundamentals of the database system.
CLO2	To give an idea how ERD components are helpful in database design and implementation.
CLO3	To enable the students use of MYSQL in the database.
CLO4	To familiarize the student with normalization, database protection and different DCL Statements.
CLO5	To make students aware about the importance of protecting data from unauthorized users.

Course Outcome

	On completing the course, the student will be able to:
CO1	Analyze database requirements and determine the entities involved in the system and their relationship to one another.
CO2	Design the database schema with the use of appropriate data types for storage of data in the database.
CO3	Create, manipulate, query and back up the databases.
CO4	Apply queries to MySQL related to String, Maths and Date Functions, views and triggers.
CO5	Evaluate the principles of normalization and assess its role in the database design process.

Detailed Syllabus

Module	Title with content	No. of lectures
I	**Introduction to DBMS –** Database, DBMS – Definition, Overview of DBMS, Advantages of DBMS, Levels of abstraction, Data independence, DBMS Architecture. **Data models** - Client/Server Architecture, Object Based Logical Model, Record Based Logical Model (relational, hierarchical, network). **Entity Relationship Model -** Entities, attributes, entity sets, relations, relationship sets, Additional constraints (key constraints, participation constraints, weak entities, aggregation / generalization, Conceptual Design using ER (entities VS attributes, Entity VS relationship, binary VS ternary, constraints beyond ER).Constraints - primary key, referential integrity, unique constraint, Null constraint, Check constraint. **ER to Table-** Entity to Table, Relationship to tables with and without key constraints. **Normal forms:** Functional dependencies, first, second, third, and BCNF normal forms based on primary keys, lossless join decomposition.	**15**
I	**DDL Statements** - Creating Databases, Using Databases, datatypes, Creating Tables (with integrity constraints – primary key, default, check, not null), Altering Tables, Renaming Tables, Dropping Tables, Truncating Tables, Backing Up and Restoring databases. **DML Statements** – Viewing the structure of a table insert, update, delete, Select all columns, specific columns, unique records, conditional select, in clause, between clause, limit, aggregate functions (count, min, max, avg, sum), group by clause, having clause. **Functions** – String Functions (concat, instr, left, right, mid, length, lcase/lower, ucase/upper, replace, strcmp, trim, ltrim, rtrim), Math Functions (abs, ceil, floor, mod, pow, sqrt, round, truncate) Date Functions (adddate, datediff, day, month, year, hour, min, sec, now, reverse). **Joins** – inner join, outer join (left outer, right outer, full outer) . **Subqueries** – subqueries with IN, EXISTS, subqueries restrictions, Nested subqueries, ANY/ALL clause, correlated subqueries. **DCL Statements** (creating/dropping users, privileges introduction, granting/revoking privileges, viewing privileges). **Views and Triggers-** (creating, altering dropping, renaming and manipulating views).Concept of triggers, Implementing triggers – creating triggers, Insert, delete, and update triggers.	**15**

References:

1. "Fundamentals of Database System", Elmasri Ramez, Navathe Shamkant, Pearson Education, Seventh edition, 2017.
2. "Database Management Systems", Raghu Ramakrishnan and Johannes Gehrke, 3rd Edition, 2014
3. "Murach's MySQL", Joel Murach, 3rd Edition, 3rd Edition, 2019

Additional References:

1. "Database System Concepts", Abraham Silberschatz, Henry F.Korth, S.Sudarshan, McGraw Hill, 2017
2. "MySQL: The Complete Reference", Vikram Vaswani , McGraw Hill, 2017
3. "Learn SQL with MySQL: Retrieve and Manipulate Data Using SQL Commands with Ease", Ashwin Pajankar, BPB Publications, 2020
4. "Beginning MySQL", Robert Sheldon, Geoff Moes, , Wrox Press, 2005.

Evaluation for Theory Courses (2 Credit Courses)

I. Continuous Internal Evaluation (CIA) - 20 Marks (40%)	**II. Semester End Theory Examination (SEE)- 30 Marks (60%)**
(i) Mid-Term Class Test – 10 Marks (ii) Assignment/ Case study/ Presentations – 10 Marks	A Semester End Theory Examination of 1 hour duration for 30 Marks as per the paper pattern given below.

Format of Theory question paper pattern

Question	**Based on**	**Options**	**Marks**
Q.1	Module I	Any 2 out of 4	10
Q.2	Module II	Any 2 out of 4	10
Q.3	Module I & II	Any 2 out of 4	10
		Total	30

- All questions shall be compulsory with internal choice within the questions.
- Each Question may be subdivided into sub questions as a, b, c, d, etc. & Each sub question carries 5 Marks.

Course/ Paper Title	**Computer Science Practical 1**
Course offered as	**Major Practical**
Course Code	**RUCSMJP1**
Semester	**I**
No. of Credits	**2 (60 hours)**
No. of lecture Hours/week	**4**

Sr No.	**Course Learning Objectives of module 1:**
CLO1	Analyze the operations of basic, universal, and general-purpose gates and verify their truth tables through experimentation.
CLO2	Implement boolean expressions using truth tables and KMaps.
CLO3	Implement and evaluate adders, subtractors and analyze SR flip flop.
CLO4	Design and verify a comparator and counter.
CLO5	Develop a comprehensive assembly language application that integrates data transfer.
CLO6	To introduce the concepts of ER diagram.
CLO7	To perform essential database operations, DDL, DML and DCL statements.
CLO8	To familiarize the concept of different queries using date, string, and mathematical functions in MYSQL.
CLO9	To compare and contrast views and tables, explore the principles of normalization, and analyze the use of triggers and joins
CLO10	To make students aware about the importance of protecting data from unauthorized users.

Course Outcome

	On completing the course, the student will be able to:
CO1	Analyze and verify truth tables for basic, universal, and general-purpose gates through practical experiments.
CO2	Implement NAND and NOR gates using basic gates and simplify Boolean expressions with truth tables and KMaps.
CO3	Design and verify the functionality of half and full adders, subtractors and flip-flops.
CO4	Construct and evaluate a 4-bit magnitude comparator, a 3-bit counter and various types of 4-bit shift registers.
CO5	Design and implement basic 8085 assembly language application.
CO6	Design and create conceptual models using ER diagrams, identifying entities, attributes, keys, and relationships.

CO7	Perform basic database operations, including creating and managing databases and tables, and manipulating records.
CO8	Execute queries using date, string, and mathematical functions to manipulate and retrieve data.
CO9	Implement and create views, triggers, joins and subqueries.
CO10	Apply DCL statements to grant and revoke permissions, understanding their implications on database security.

Modules	**Module 1 (30Hrs) & Module 2 (30Hrs)**
Module 1	**Digital Systems - Practical**
Sr No.	**Experiments**
1	Study and verify the truth table of the following: a) Basic Gates - NOT, AND, OR b) Universal Gates - NAND, NOR c) General purpose Gates- EX-OR, and EX-NOR
2	Implement NAND and NOR Gates using Basic Gates.
3	Implement Circuit diagram of given Boolean expression and draw truth table and KMap.
4	Design and verify a half/ full Adders and Subtractors.
5	Design and verify the operation of SR & D flip-flops using logic gates.
6	Design a 4 bit Magnitude Comparator using combinational circuits.
7	Verify the operation of a 3 bit Counter.
8	Verify the operation of a 4 bit shift register. Implement any one from SISO, SIPO, PISO, PIPO shift register.
9	Design and implement expressions using multiplexers / demultiplexers.
10	a. Write an assembly language program to perform data transfer instructions. operations. b. Write an assembly language program to accept n numbers and display the sum of n numbers. stop accepting numbers until you enter zero.
	Note: Practical 01 – 09 can be performed using Logisim software (Download it from https://sourceforge.net/projects/circuit/) Practical 10 can be performed on any simulation software like 8085 simulator

Module 2	Database Management System - Practical
Sr No.	**Experiments**
1	Conceptual Designing using ER Diagrams (Identifying entities, attributes, keys and relationships between entities, cardinalities, generalization, specialization etc.)
2	Perform the following: a) Viewing all databases b) Creating a Database c) Viewing all Tables in a Database d) Creating Tables (With and Without Constraints) Inserting/Updating/Deleting Records in a Table
3	Perform the following: a) Altering a Table b) Dropping/Truncating/Renaming Tables Backing up / Restoring a Database
4	Perform the following: a) Simple Queries b) Simple Queries with Aggregate functions Queries with Aggregate functions (group by and having clause)
5	Queries involving a) Date Functions b) String Functions c)Math Functions
6	Join Queries a) Inner Join b)Outer Join
7	Subqueries a) With IN clause b)With EXISTS clause
8	DCL statements- Granting and revoking permissions
9	Converting ER Model to Relational Model and apply Normalization on database. (Represent entities and relationships in Tabular form, Represent attributes as columns, identifying keys and normalization up to 3rd Normal Form).
10	Views -a) Creating Views (with and without check option) b)Dropping views c) Selecting from a view Triggers:-Creating and working with Insert/Update/Delete Trigger using Before/After clause.

Practical Evaluation (50 Marks)

Sr. No.	Practical Assessment	Marks
1	Practical Question 01 from module 1	15
2	Practical Question 02 from module 2	15
3	Viva	10
4	Journal	10

- Duration of Practical exam is of 02 Hrs.
- Certified Journal is compulsory for appearing at the time of practical Exam.
- Minimum 80% practical from each module are required to be completed.
- The total Evaluation is out of 50 marks.

Course/ Paper Title	**Discrete Mathematics**
Course offered as	**Minor**
Course Code	**RUCSMN103**
Semester	**I**
No. of Credits	**2 (30 hours)**
No. of lecture Hours/week	**2**

Sr No.	Course Learning Objectives of module 1:
CLO1	To introduce students to foundational concepts in Discrete Mathematics.
CLO2	To develop analytical skills necessary for problem-solving in various fields of computer science, mathematics, and beyond.
CLO3	To introduce functions, forming and solving recurrence relations and different counting principles.
CLO4	To Enable the students to solve problems based on graphs and trees, which are widely used in software.

Course Outcome

	On completing the course, the student will be able to:
CO1	Define mathematical structures (relations, functions, graphs) and use them to model real life situations.
CO2	Construct and solve simple mathematical problems.
CO3	Explain the concept of Graphs and Trees.
CO4	Apply the concepts of various types of relations, partial ordering and equivalence relations.
CO5	To differentiate functions and solve recurrence relations problems.

CO6	Apply the concepts of counting and probability

Detailed Syllabus

Module	Title with content	No. of lectures
I	**Introduction:** Variables, The Language of Sets, The Language of Relations and Function, Set Theory: Definitions and the Element Method of Proof, Properties of Sets. **Functions:** Definition of function. Domain, co domain and the range of a function. Direct and inverse images. Injective, surjective and bijective functions. Composite and inverse functions. **Relations:** Relations on Sets, Reflexivity, Symmetry, and Transitivity, Equivalence Relations, Partial Order Relations, Hasse Diagram. **Recurrence Relations:** Definition of recurrence relations, Formulating recurrence relations, solving recurrence relations- Back tracking method**,** Applications- Formulate and solve recurrence relation for Fibonacci numbers, Tower of Hanoi, Intersection of lines in a plane, Sorting Algorithms.	**15**
II	**Permutations and Combinations:** Partition and Distribution of objects, Permutation with distinct and indistinct objects, Counting Principles: Sum and Product Rules, Two-way counting, Tree diagram for solving counting problems, Pigeonhole Principle (without II proof); Simple examples, Inclusion Exclusion Principle (Sieve formula) (Without proof). **Graphs:** Definitions and Basic Properties, Trails, Paths, and Circuits, Matrix Representations of Graphs, Operations on graph with algorithms - searching in a graph; Insertion in a graph, Deleting from a graph, Traversing a graph- Breadth-First search and Depth-First search. **Trees:** Definition and elementary results. Ordered rooted tree, Binary trees, Complete and extended binary trees, representing binary trees in memory, traversing binary trees, binary search tree, Algorithms for searching and inserting in binary search trees, Algorithms for deleting in a binary search tree.	**15**

References:

1. Discrete Mathematics and Its Applications, Seventh Edition by Kenneth H. Rosen, McGraw Hill Education (India) Private Limited. (2011)
2. Discrete Mathematics: Seymour Lipschutz, Marc Lipson, Schaum's out lines, McGraw- Hill Inc.3rd Edition
3. Data Structures Seymour Lipschutz, Schaum's out lines, McGraw- Hill Inc. 2017
4. Norman L. Biggs, Discrete Mathematics, Revised Edition, Clarendon Press, Oxford 1989.

Additional References:

1. Elements of Discrete Mathematics: C.L. Liu, Tata McGraw- Hill Edition.
2. Concrete Mathematics (Foundation for Computer Science): Graham, Knuth, Patashnik Second Edition, Pearson Education.
3. Discrete Mathematics: Semyour Lipschutz, Marc Lipson, Schaum's out lines, McGraw- Hill Inc.
4. Foundations in Discrete Mathematics: K.D. Joshi, New Age Publication, New Delhi.

Evaluation for Theory Courses (2 Credit Courses)

I. Continuous Internal Evaluation (CIA) - 20 Marks (40%)	II. Semester End Theory Examination (SEE)- 30 Marks (60%)

(iii) Mid-Term Class Test – 10 Marks (iv)Assignment/ Case study/ Presentations – 10 Marks	A Semester End Theory Examination of 1 hour duration for 30 Marks as per the paper pattern given below.

Format of Theory question paper pattern

Question	Based on	Options	Marks
Q.1	Module I	Any 2 out of 4	10
Q.2	Module II	Any 2 out of 4	10
Q.3	Module I & II	Any 2 out of 4	10
		Total	30

- All questions shall be compulsory with internal choice within the questions.
- Each Question may be subdivided into sub questions as a, b, c, d, etc. & Each sub question carries 5 Marks.

Course/ Paper Title	**Linux Programming**
Course offered as	**VSC**
Course Code	**RUCSVSC106**
Semester	**I**
No. of Credits	**2** (60 hours)
No. of lecture Hours/week	**4**

Sr No.	Course Learning Objectives:
CLO1	Demonstrate the process of installing and configuring the Ubuntu Linux operating system and its software packages.
CLO2	Utilize essential file systems and handling commands to manage files and directories effectively.
CLO3	Apply general-purpose utility and networking commands to enhance system interaction and connectivity.
CLO4	Apply simple filters and I/O redirection techniques to manipulate and process data efficiently in a Linux environment.
	Utilize text editors to edit files, manage processes effectively, and create shell scripts

Course Outcome

	On completing the course, the student will be able to:
CO1	Install and configure the Ubuntu operating system.
CO2	Execute commands for file creation, navigation, and deletion in the file system.
CO3	Use networking commands to establish connections and gather system information.
CO4	Create and modify scripts using editors and apply filters for efficient text manipulation.
CO5	Apply essential commands in text editors and process management.
CO6	Install and configure the Ubuntu operating system.

	Linux Programming - Practical
Sr No.	**Experiments**
1	Installation of Ubuntu Linux operating system. a. Booting and Installing from (USB/DVD) b. Using Ubuntu Software center c. Change your Desktop and Lock Screen Background.
2	Becoming an Ubuntu power user a. Administering system and User setting b. Using the Terminal c. Operations performed on Linux Operating System: Screen Resolution, Networking, Time settings.
3	Installing and Removing Software: Install gcc package. Verify that it runs, and then remove it.
4	File System Commands: touch, help, man, more, less, pwd, cd, mkdir, rmdir, ls, find, etc
5	Documentations I: a. Finding Info Documentation: From the command line: bring up the info page for the grep command . Bring up the usage section. b. Finding man pages by Topic What man pages are available that document file compression ?
7	Command line operations I: a. Install any new package on your system. b. Remove the package installed. c. Find the passwd file in / using find command. d. Create a symbolic link to the file you found in last step.
8	General Purpose Utility Commands:cal, date, echo, man, printf, passwd, script, who, uname, tty, stty, etc
9	Networking Commands: who, whoami, ping, telnet, ftp, ssh, etc.
10	File Operations I: a. Explore mounted file systems on your system. b. Archive and backup your home directory or work directory using tar, gzip commands.
11	File Operations II: a. What are different ways of exploring mounted file systems on Linux? b. Use dd command to create files and explore different options to dd.
12	Editors: vi, sed, awk
13	Use environment I a) Which account are you logged in? How do you find out? b) Display /etc/shadow file using cat and understand the importance of shadow file. How it‘s different than passwd file.
14	Use environment II a) Explore different ways of getting command history, how to run previously executed command without typing it? b) Get your current working directory.

15	Linux Editors: vim/gedit/nano a. Create, modify, search, navigate a file in editor. b. Learn all essential commands like search, search/replace, highlight, show line numbers.
16	Working and managing with processes- sh, ps, kill, nice, at, batch, etc.
17	Shell scripting, I: Defining variables, reading user input, exit and exit status commands, expr, test, [], if conditional, logical operators
18	Shell scripting II: Conditions (for loop, until loop and while loop) arithmetic operations, examples
19	Simple Filters and I/O Redirection: head, tail, cut, paste, sort, grep family, tee, uniq, tr,etc.
20	File handling Commands: cat, cp, rm, mv, more, file, wc, od, cmp, diff, comm, chmod, chown, chgrp, gzip and gunzip, zip and unzip, tar, ln, umask, chmod, chgrp, chown, etc

Practical Evaluation (50 Marks)

Sr. No.	Practical Assessment	Marks
1	Practical Question 01	20
2	Practical Question 02	20
3	Viva	05
4	Journal	05

- Duration will be 02 hrs. for each practical course
- Certified Journal is compulsory for appearing at the time of practical Exam.
- Minimum 80% practical from each core subject are required to be completed.

Course/ Paper Title	**Programming with Python**
Course offered as	**SEC**
Course Code	**RUCSSEC107**
Semester	**I**
No. of Credits	**2 (60 hours)**
No. of lecture Hours/week	**4**

Sr No.	Course Learning Objectives:
CLO1	To explore and understand the fundamental concepts and principles of Python programming
CLO2	To examine and discuss the pros and cons of scripting languages compared to classical programming languages through practical implementation.
CLO3	To develop the ability to write basic Python scripts and apply essential programming techniques.
CLO4	To investigate Python's core libraries and apply basic functions in the creation of meaningful programs.
CLO5	To familiarize students with Python's IDLE (Integrated Development and Learning Environment) for effective coding practices.

CLO6	To cultivate logical thinking and enhance problem-solving skills by engaging with fundamental programming concepts.

Course Outcome

	On completing the course, the student will be able to:
CO1	Introduce the fundamental concepts and principles of Python programming.
CO2	Analyze the advantages and disadvantages of scripting languages compared to classical programming languages by implementing various control statements.
CO3	Apply basic skills to write Python scripts effectively.
CO4	Evaluate Python's core libraries and utilize basic functions to develop meaningful programs.
CO5	Demonstrate proficiency in using Python's IDLE
CO6	Enhance logical thinking and develop problem-solving skills by introducing fundamental programming concepts.

	Python Programming - Practical
Sr No.	**Experiments**
1	Installation of python Software, Understanding of various variants of python editor, and performing basic use of Shell and comments.
2	Write a Python program to check data types of the variables, use keywords and perform type casting.
3	Write a Python program to demonstrate the use of operators, precedence and associativity of operators.
4	Write a Python program to calculate the area and perimeter of a rectangle, Square and Circle.
5	A. Write a program to swap two numbers. B. Write a program to swap three numbers without using Temporary variables.
7	A. Write a program to find the largest of three numbers. B. Write a program to check whether the entered year is a leap year or not. C. Write a program to check whether the entered number is even or odd. D. Write a program to check whether the entered number is positive, negative or zero.
8	A. Write a program to check whether the entered character is a vowel or not. B. Write a program to check whether the entered character is an upper case or a lower case.
9	Write a program that takes the marks of 5 subjects and displays the grade.
10	A. Write a program to find the reverse of a number. B. Write a program to check if the entered string is palindrome or not. C. Write a program to check whether the entered number is an Armstong number or not. D. Write a program to check whether the entered number is a Strong number or not.
11	A. Write a program to find the Factorial of a number. B. Write a program to print the fibonacci series of n numbers. C. Write a program to solve the Tower of Hanoi Problem. D. Write a program to print the first 20 prime numbers.
12	Write a program to print various types of patterns in python.

13	Write a program to demonstrate the use of A. Break statement B. Continue statement C. Pass statement D. Return statement.
14	Write a program to demonstrate different types of functions.
15	Write a python program to perform the following operations on sets: the intersection of sets, the union of sets, set difference, symmetric difference,clear a set.
16	Write a program to implement tuples in Python for suitable problems. Demonstrate various operations on it.
17	Write a program to implement a list in Python for suitable problems. Demonstrate various functions & operations on it.
18	Write a program to implement a dictionary in Python for suitable problems. Demonstrate various functions & operations on it.
19	Write a program to create and Manipulate Strings by using various built in methods.
20	Write a program to create and manipulate arrays in Python and perform different operations on it.

Practical Evaluation (50 Marks)

Sr. No.	Practical Assessment	Marks
1	Practical Question 01	20
2	Practical Question 02	20
3	Viva	05
4	Journal	05

- Duration will be 02 hrs. for each practical course
- Certified Journal is compulsory for appearing at the time of practical Exam.
- Minimum 80% practical from each core subject are required to be completed.

Course/ Paper Title	**Communication Skills**
Course offered as	**AEC**
Course Code	**RUCSAEC108**
Semester	**I**
No. of Credits	**2 (30 hours)**
No. of lecture Hours/week	**2**

Sr No.	Course Learning Objectives:
CLO1	State the importance of communication skills and various types of communication.
CLO2	Describe effective verbal and non-verbal communication through group discussions (GD), presentations (PI), and professional settings.
CLO3	State the role of communication in collaborative work and how it impacts teamwork and relationships.

CLO4	Explain different communication strategies and their effectiveness in various contexts.
CLO5	Demonstrate the professional presentation that incorporates best practices in communication

Course Outcome

	On completing the course, the student will be able to:
CO1	Use communication skills effectively.
CO2	Organize knowledge, skills and judgment around human communication that facilitate their ability to work collaboratively with others.
CO3	Demonstrate the importance and types of communications skills.
CO4	Using verbal and non- verbal Communication using GD, PI and professional presentation.
CO5	Formulate effective Communication skills through presentations.

Detailed Syllabus

Module	Title with content	No. of lectures
I	**Components of effective communication:** seven cs of effective communication, Nature and Scope of Communication, Composing effective messages, Verbal Communication & Non – Verbal Communication: its importance and nuances: Facial Expression, Posture,Gesture, Eye contact, appearance (dress code), Barriers to Communication. **Listening Skills:** Purpose of Listening , Listening to Conversation (Formal and Informal) , Active Listening- an Effective Listening Skill , Benefits of Effective Listening , Barriers to Listening , Listening to Announcements- (railway/ bus stations/ airport /sports announcement/ commentaries etc.) **Personality Development**: Knowing Yourself, Positive Thinking, Johari‟s Window, Physical Fitness. **Employment Communication:** Introduction, Resume, Curriculum Vitae,Scannable Resume, Developing an Impressive Resume, Formats of Resume, Job Application or Cover Letter	15
II	**Job Interviews:** Introduction, Importance of Resume, Definition of Interview, Background Information, Types of Interviews, Preparatory Steps,for Job Interviews, Interview Skill Tips, Changes in the Interview Process,FAQ During Interviews **Group Discussion:** Introduction, Ambience/Seating Arrangement for Group Discussion, Importance of Group Discussions, Difference between Group Discussion, Panel Discussion and Debate, Traits, Types of Group Discussions, topic based and Case based Group Discussion, Individual Traits **Understanding Specific Communication Needs:** Corporate Communication, Persuasive Strategies in Business Communication, Ethics in Business Communication, Business Communication Aid	15

References:

1. Soft Skills: An Integrated Approach to Maximize Personality, Gajendra S. Chauhan, Sangeeta Sharma, Wile India

2. Strategies for improving your business communication by Prof. M. S. Rao Prakash Singh, Shroff Publisher and distributors

Additional References:

1. Communication Skills. Pune: Technical Publications, by Mathew, S.
2. Personality Development and Soft Skills, Barun K. Mitra, Oxford Press

Evaluation Pattern (2 Credit Courses) (50 marks)

- **Class Test (25 marks)**
- **Project/Assignment/Case Study (20 Marks)**
- **Attendance (05 marks)**

Course/ Paper Title	**Human Values**
Course offered as	**VEC**
Course Code	**RUCSVEC109**
Semester	**I**
No. of Credits	**2 (30 Hours)**
No. of lecture Hours/week	**2**

Sr No.	**Course Learning Objectives:**
CLO1	Foster personal, familial, societal, and environmental development to instill positive values in professional relationships.
CLO2	Differentiate between the values and skills needed for success in professional and personal environments.
CLO3	Articulate the core values essential for achieving excellence in one's career.
CLO4	Design a comprehensive strategy for cultivating long-term human relationships based on trust and understanding.
CLO5	Analyze global issues related to energy scarcity and develop effective solutions to address these challenges.

Course Outcome

	On completing the course, the student will be able to:
CO1	Evaluate the impact of personal, familial, societal, and environmental factors on professional relationships.
CO2	Compare and contrast the values and skills necessary for success in both professional and personal settings.
CO3	Explain the core values that contribute to career excellence and their importance in various professional contexts.
CO4	Create a detailed plan to nurture long-term relationships grounded in trust and mutual understanding.
CO5	Assess global energy scarcity issues and propose innovative strategies to mitigate their effects.

Detailed Syllabus

Module	Title with content	No. of lectures
I	**Introduction to Professional Ethics:** Basic Concepts, Governing Ethics, Personal and Professional Ethics, Ethical Dilemmas, Life Skills, Emotional Intelligence, Thoughts of Ethics, Value Education, Dimensions of Ethics. **Profession and professionalism**: Introduction, Professional Associations, Professional Risks, Professional Accountabilities, Professional Success, Ethics and Profession. **Basic Theories:** Basic Ethical Principles, Moral Developments, Deontology, Utilitarianism, Virtue Theory, Rights Theory, Casuist Theory, Moral Absolution, Moral Rationalism, Moral Pluralism , Ethical Egoism, Feminist Consequentialism, Moral Issues, Moral Dilemmas, Moral Autonomy.	**15**
II	**Ethics in Engineering Profession**: Introduction, Technology and Society, Engineering Professions and Norms of Professional Conduct, Engineering ethics, Issues in Engineering ethics, Role of Engineers, Engineers as Managers, other roles played by engineers. **Engineers Responsibilities For safety**: Safety and risk, assessment of risk, Safety in engineering products, risk and cost, Designing for safety, Risk-Benefit Analysis,Risk costs and management,Principles of Risk Management.Strategies of risk management. **Responsibilities and rights of professionals**: Professional responsibilities, Professional rights. **Global issues in Professional ethics:** Introduction – Current Scenario, Technology, Globalization of MNCs, International Trade, World Summits, Issues,Business Ethics and Corporate Governance. **Sustainable Development Ecosystem**, Energy Concerns, Ozone Deflection, Pollution, Ethics in Manufacturing, Ethics in Marketing, Global Issues, Ethics in Private and Public Relationships, The impact of technology in Quality of life.	**15**

References:

1. R. Subramanian, Professional Ethics - Includes Human Values, Oxford University Press, Second Edition 2017
2. R.S. Naagarazan, Professional Ethics and Human Values, New Age International (P) Limited Publishers,2006

Additional References:

1. Joe Edelman ,Human Values: A Quick Primer

Evaluation Pattern (2 Credit Courses) (50 marks)

- **Class Test (25 marks)**
- **Project/Assignment/Case Study (20 Marks)**
- **Attendance (05 marks)**

Course/ Paper Title	**Development in Science and Technology**
Course offered as	**IKS: Indian Knowledge System**
Course Code	**RUCSIKS110**
Semester	**I**
No. of Credits	**2 (30 hours)**
No. of lecture Hours/week	**2**

Sr No.	**Course Learning Objectives:**
CLO1	To familiarize learners with major sequential development in Indian science and technology.
CLO2	To make the students understand the stages of scientific methods.
CLO3	Describe Indian traditional practices, nanotechnology, meteorology, Textile Technology and Metallurgy in India

Course Outcome

	On completing the course, the student will be able to:
CO1	Describe major sequential developments in Indian science and technology.
CO2	Demonstrate an understanding of the stages of the scientific method and superstition.
CO3	Demonstrate an understanding of ancient Indian Practices in science and technology.
CO4	Explain the developments in Nanotechnology Metallurgy, Textile Technology, Metallurgy and Meteorological science.

Detailed Syllabus

Module	**Title with content**	**No. of lectures**
I	**Introduction to Science and Technology:** Science and Technology: Introduction, Role of Science and Technology in today's life, Nature and Scope.Fundamental Concepts in Scientific Thinking, Development of Science and Technology in India, Science and technology policies in India, Impact of Science and Technology on Society.. **Stages in Scientific Methods:** different stages in the Study of Science Observation, Research, Hypothesis , Experiment, Analysis, Result. Science and Superstitions. **Indian Traditional Knowledge in science & Technology practices :** Traditional water-harvesting practices, Traditional Livestock and veterinary Sciences Traditional Houses & villages, Traditional Forecasting, Traditional Ayurveda & plant based medicine, Traditional writing Technology.	**15**

II	**Development of science in Nanotechnology:** Introduction of Nanotechnology , Application of Nanotechnology in Different Fields, Nano-medincine , Advantages and Disadvantages of Nanomedicine. **Meteorological science:** Introduction to meteorology, six main components of the weather, compare Meteorology with Climatology . **Textile Technology in India:** Cotton (natural cellulose fiber), silk, wool (natural protein fibers), needle and thread – tools for stitching. dyeing, washing spinning and weaving technology, **Metallurgy in India:** Survarṇa(gold) and its different types, prosperities, Rajata(silver), Tamra(copper), Loha(iron), Vanga(tin), Naga / sisa(lead), Pittala(brass)	**15**

References:

1. Textbook on IKS by Prof. B Mahadevan, IIM Bengaluru.
2. Kapur K and Singh A.K (Eds) 2005). Indian Knowledge Systems, Vol. 1. Indian Institute of Advanced Study, Shimla. Tatvabodh of sankaracharya, Central chinmay mission trust, Bombay, 1995.
3. Nair, Shantha N. Echoes of Ancient Indian Wisdom. New Delhi: Hindology Books, 2008.

Additional References:

1. SK Das, The education system of Ancient hindus, Gyan publication house, India
2. R P Kulkarni, Glimpese of Indian Engineering and Technology (Ancient & Medieval period, Munshiram Manoharlal Publishers Pvt. Ltd. 2018
3. Annual Review of Information Science and Technology (ARIST) 39. By Blaise Cronin, Information Today, 2004.

Evaluation Pattern (2 Credit Courses) (50 marks)

- **Class Test (25 marks)**
- **Project/Assignment/Case Study (20 Marks)**
- **Attendance (05 marks)**

SEMESTER II

Course/ Paper Title	**Object Oriented Programming**
Course offered as	**Major I**
Course Code	**RUCSMJ201**
Semester	**II**
No. of Credits	**2 (30 hours)**
No. of lecture Hours/week	**2**

Sr No.	**Course Learning Objectives:**
CLO1	Explain the core object oriented and procedure concepts and apply OOP principles to create efficient C++ code.
CLO2	Identify and explain the features of C++, data types and the process of compiling and running C++ programs.
CLO3	Demonstrate decision-making and looping statements for problem-solving.
CLO4	Analyze and differentiate between various types of inheritance
CLO5	Explain and implement polymorphism and memory management in C++.
CLO6	Demonstrate file handling in C++ using file stream classes

Course Outcome

	On completing the course, the student will be able to:
CO1	Compare the key differences between procedure-oriented programming and object-oriented programming.
CO2	Demonstrate the basic structure and syntax of a C++ program, including data types and variable naming, and compile/run simple programs.
CO3	Apply decision-making & looping statements to create programs.
CO4	Implement classes in C++ and use constructors, destructors, and member functions effectively.
CO5	Develop programs that utilize both compile-time and runtime polymorphism through function and operator overloading.
CO6	Analyze and implement inheritance, pointer usage and file handling techniques.

Detailed Syllabus

Module	**Title with content**	**No. of lectures**
I	**Basics of OOPS:** Object oriented programming paradigm, basic concepts of object oriented programming,comparing Procedure oriented programming with object oriented programming , benefits of object oriented programming, object oriented languages, applications of object oriented programming. **Introduction of C++:** Introduction and features of C++, Basic structure of C++ program , data types,rules for naming variables, keywords , identifiers,Tokens. **Programming Concepts**: simple C++ program without classes, cout and cin statements in C++ ,compiling and running C++ program. programming with classes, Defining constants.	**15**

	Operators in C++: operators, precedence of operators, type cast operator, implicit and explicit type casting, conditional operator. **Decision Making, Loops, Arrays and Strings**: Conditional statements-if, if...else, switch loops- while, do...while, for, types of arrays and string and string manipulations. **Classes, Abstraction & Encapsulation:** Classes and objects, Dot Operator, data members, member functions, passing data to functions,scope and visibility of variables in function.	
II	**Constructors and Destructors:** Default constructor, parameterized constructor, copy constructor, destructors. Static members - static data and static function, access specifiers. **Polymorphism -** Compile time polymorphism - function overloading & operator overloading , constructor overloading , Runtime Polymorphism :function overriding. **Inheritance**: Defining base class and its derived class, access specifiers,types of inheritance-single, multiple, hierarchical, multilevel, hybrid inheritance. **Virtual function:** Virtual function, pure virtual function, virtual base class, abstract class. **Pointers**: Introduction to pointers, * and & operators, assigning addresses to pointer variables, accessing values using pointers, pointers to objects & this pointer, pointers to derived classes **File Handling**: File Stream classes, opening and closing file-file opening modes, text file handling, binary file handling.	**15**

References:

1. Object Oriented Programming with C++, Balagurusamy E., 8th Edition, McGraw Hill Education India.
2. Let Us C++ by KanetkarYashwant, Publisher: BPB Publications, 2020

Additional References:

1. Mastering C++ by Venugopal, Publisher: McGraw-Hill Education, 2017
2. Object Oriented Analysis and Design by Timothy Budd TMH, 2001

Evaluation for Theory Courses (2 Credit Courses)

I. Continuous Internal Evaluation(CIA) - 20 Marks (40%)	**II. Semester End Theory Examination (SEE)- 30 Marks (60%)**
(i) Mid-Term Class Test – 10 Marks (ii) Assignment/ Case study/ Presentations – 10 Marks	A Semester End Theory Examination of 1 hour duration for 30 Marks as per the paper pattern given below.

Format of Theory question paper pattern

Question	Based on	Options	Marks
Q.1	Module I	Any 2 out of 4	10
Q.2	Module II	Any 2 out of 4	10
Q.3	Module I & II	Any 2 out of 4	10
Total			30

- All questions shall be compulsory with internal choice within the questions.
- Each Question may be subdivided into sub questions as a, b, c, d, etc. & Each sub question carries 5 Marks.

Course/ Paper Title	**Design & Analysis of Algorithm**
Course offered as	**Major II**
Course Code	**RUCSMJ202**
Semester	**II**
No. of Credits	**2 (30 hours)**
No. of lecture Hours/week	**2**

Sr No.	Course Learning Objectives:
CLO1	To introduce the students to design and analyze algorithms of efficiency and correctness.
CLO2	Compare different algorithms based on their running time and rate of growth.
CLO3	Apply and Implement searching techniques and sorting algorithms.
CLO4	Identify and differentiate various data structures.
CLO5	Enable students to understand and apply key algorithm design paradigms for effectively solving complex computational problems.

Course Outcome

	On completing the course, the student will be able to:
CO1	Define basic concepts of program development, algorithms and flowchart.
CO2	Solve problems by developing programs in Python.
CO3	Design algorithms by selecting the best techniques.
CO4	Compare data structures and choose the best one for a design scenario.
CO5	Explain and apply key algorithmic design paradigms.

Detailed Syllabus

Module	Title with content	No. of lectures
I	**Introduction to algorithms** - What is algorithm?, Analysis of algorithm, Types of complexity, Running time analysis, How to Compare Algorithms, Rate of Growth, Types of Analysis, Asymptotic Notation, Big-O Notation, Omega-Ω Notation, Theta-Θ Notation, Asymptotic Analysis, Performance characteristics of algorithms, Estimating running time, number of steps of executions on paper, Idea of Computability **Introduction to Data Structures** - What is data structure?, types, Introduction to Array(1-d & 2-d), Stack and List data structures, operations on these data structures, advantages disadvantages and applications of these data structures like solving linear equations, Polynomial Representation, Infix-to-Postfix conversion	**15**

	Recursion - What is recursion, Recursion vs Iteration, recursion applications like Factorial of a number, Fibonacci series & their comparative analysis with respect to iterative version, Tower of hanoi problem **Basic Searching & Sorting Techniques -** Types of Searching**:** Linear Search, Binary Search and their comparative analysis, Types of Sorting: Bubble, Selection, Insertion Sort & their comparative analysis.	
II	**String Algorithms -** Pattern matching in strings, Brute Force Method & their comparative analysis **Greedy Technique -** Concept, Advantages & Disadvantages, Applications, Implementation using problems like - file merging problem **Divide-n-Conquer -** Concept, Advantages & Disadvantages, Applications, Implementation using problems like - merge sort, Strassen's Matrix Multiplication **Dynamic Programming -** Concept, Advantages & Disadvantages, Applications, Implementation using problems like - Fibonacci series, Factorial of a number, Longest Common subsequence **Backtracking Programming -** Concept, Advantages & Disadvantages, Applications, Implementation using problems like N-Queen Problem	**15**

References:

1. "Data Structure and Algorithm Using Python", Rance D. Necaise, Wiley India Edition, 2016.
2. "Data Structures and Algorithms Made Easy", Narasimha Karumanchi, Career Monk Publications, 2016.
3."Introduction to Algorithms", Thomas H. Cormen, 3rd Edition, PHI.

Additional References:

1. "Introduction to the Design and Analysis of Algorithms", Anany Levitin, Pearson, 3rd Edition, 2011.
2. "Design and Analysis of Algorithms", S. Sridhar, Oxford University Press, 2014

I. Continuous Internal Evaluation(CIA) - 20 Marks (40%)	**II. Semester End Theory Examination (SEE)- 30 Marks (60%)**
(iii)Mid-Term Class Test – 10 Marks (iv)Assignment/ Case study/ Presentations – 10 Marks	A Semester End Theory Examination of 1 hour duration for 30 Marks as per the paper pattern given below.

Format of Theory question paper pattern

Question	Based on	Options	Marks
Q.1	Module I	Any 2 out of 4	10
Q.2	Module II	Any 2 out of 4	10
Q.3	Module I & II	Any 2 out of 4	10
Total			30

- All questions shall be compulsory with internal choice within the questions.
- Each Question may be subdivided into sub questions as a, b, c, d, etc. & Each sub question carries 5 Marks.

Course/ Paper Title	**Computer Science Practical 2**
Course offered as	**Major Practical**
Course Code	**RUCSMJP2**
Semester	**II**
No. of Credits	**2** (60 hours)
No. of lecture Hours/week	**4**

Sr No.	Course Learning Objectives:
CLO1	Implement basic C++ concepts, conditional statements and loops.
CLO2	List different types arrays for data storage.
CLO3	Explain the use of constructors and destructors
CLO4	Applying function and operator overloading to enhance functionality.
CLO5	Design class structures for code reusability using inheritance.
CLO6	Apply pointers for memory management and file I/O operations.
CLO7	Describe the properties of arrays and implement various array operations in a programming environment.
CLO8	Analyze the performance of searching algorithms, including time and space complexities, and apply them to solve practical problems.
CLO9	Evaluate different sorting techniques, comparing their efficiency and selecting appropriate methods for specific data sets.
CLO10	Utilize stack operations to manage function calls and solve problems that require backtracking or depth-first search.
CLO11	Apply dynamic programming and greedy algorithms to formulate solutions for optimization problems, justifying their approach through analysis.

Course Outcome

	On completing the course, the student will be able to:
CO1	Write C++ basic programming with conditional statements and looping structures.
CO2	Demonstrate effective manipulation of one- and two-dimensional arrays using appropriate indexing.
CO3	Design and use constructors, destructors, function and operator overloading.
CO4	Implement single, multilevel, and hybrid inheritance and polymorphism.
CO5	Apply pointers for dynamic memory management and file handling operations.
CO6	Demonstrate understanding of arrays, implementing basic operations such as insertion, deletion, and traversal.

CO7	Implement different searching algorithms and evaluate their efficiency in various scenarios.
CO8	Evaluate sorting algorithms based on time complexity and space requirements.
CO9	Implement problems involving recursion and backtracking.
CO10	Apply dynamic programming and greedy algorithms and synthesize solutions from brute force approaches.

Modules	**Module 1 (30Hrs) & Module 2 (30Hrs)**
Module 1	**Object Oriented Programming – Practical**
Sr No.	**Experiments**
1	Write a program to create a C++ program without classes.
2	Write a Program to demonstrate the use of data members & member functions in classes and use of scope resolution operator.
3	Programs based on branching and conditional statements using classes.
4	Programs based on different types of looping statements using classes.
5	Program to demonstrate one and two dimensional arrays using classes.
6	Programs to demonstrate various types of constructors and destructors.
7	Programs to demonstrate function overloading and operator overloading.
8	Programs to demonstrate the following: a. single and multilevel inheritance b. Hybrid inheritance
9	Programs to display the value of variables using pointers.
10	Write a Program to read and write from the text file
Module 2	**Design & Analysis of Algorithm – Practical**
1	Programs on 1-D Arrays like – a.sum of elements of array, b.searching an element in array, c.finding minimum and maximum element in array, d.count the number of even and odd numbers in array. For all such programs, also find the time complexity.
2	Programs on 2-D Arrays like a.row-sum b.column-sum c.sum of diagonal elements d.addition of two matrices e.multiplication of two matrices. For all such programs, also find the time complexity.
3	Program to create a list-based stack and perform various stack operations.

4	Program to perform linear and binary search on list of elements.
5	Programs to sort elements of list by using various algorithms like ● bubble ● selection sort ● insertion sort. Compare the efficiency of algorithms.
6	Programs to select the Nth Max/Min element in a list by using various algorithms. Compare the efficiency of algorithms.
7	Programs to find a pattern in a given string - general way and brute force technique. Compare the efficiency of algorithms
8	Programs on recursion like factorial,Fibonacci, Tower of hanoi. Compare algorithms to find factorial/fibonacci using iterative and recursive approaches.
9	Program to implement file merging or coin change problems using Greedy Algorithm and to understand time complexity.
10	Program to implement Longest Common Subsequence using dynamic programming and to understand time complexity. Compare it with the general recursive algorithm

Practical Evaluation (50 Marks) -

Sr. No.	Practical Assessment	Marks
1	Practical Question 01 module 1	15
2	Practical Question 02 module 2	15
3	Viva	10
4	Journal	10

- Duration is of 02 Hrs .
- Certified Journal is compulsory for appearing at the time of practical Exam.
- Minimum 80% practical from each module are required to be completed.
- The total Evaluation is out of 50 marks

Course/ Paper Title	**Descriptive Statistics**
Course offered as	**Minor**
Course Code	**RUCSMN203**
Semester	**II**
No. of Credits	**2** (30 hours)
No. of lecture Hours/week	**2**

Sr No.	Course Learning Objectives:
CLO1	To introduce the different types of data.
CLO2	To equip students with the skills to effectively present data using descriptive statistics, including the ability to summarize and visualize data through tables, charts, and graphs.
CLO3	Enable students to use various measures of central tendency and dispersion in relevant contexts.
CLO4	To identify correlations between variables and apply regression analysis to determine the nature of their relationships.

Course Outcome

	On completing the course, the student will be able to:
CO1	Analyze statistical data utilizing measures of central tendency and dispersion.
CO2	Apply basic statistical techniques for data analysis.
CO3	Explain the concept of skewness and kurtosis.
CO4	Calculate and interpret the correlation between two variables.
CO5	Calculate the simple linear regression equation for a set of data.
CO6	Explain the principles of linear regression and correlation.

Detailed Syllabus

Module	Title with content	No. of lectures
I	**Data Types and Data Presentation**: Data types: Attribute, Variable, Discrete and Continuous variable, Univariate and Bivariate distribution. **Data presentation**: Frequency distribution, Histogram, Ogive curves. **Measures of Central tendency**: Concept of average/central tendency, characteristics of good measure of central tendency. Arithmetic Mean, Median, Mode. Definition, examples for ungrouped and grouped data, Combined arithmetic mean. Partition Values: Quartiles, Deciles and Percentiles. **Measures dispersion**: Concept of dispersion, characteristics of good measure of dispersion. Range, Semi-inter quartile range, Quartile deviation, Standard deviation - Definition, examples for ungrouped and grouped data, Variance and Coefficient of variation.	**15**

II	**Moments**: Concept of Moments, Raw moments, Central moments, Relation between raw and central moments **Measures of Skewness and Kurtosis**: Concept of Skewness and Kurtosis, measures based on moments, quartiles. **Correlation**: Concept of correlation, Types and interpretation, Measure of Correlation: Scatter diagram and interpretation; Karl Pearson's coefficient of correlation (r): Definition, examples for ungrouped and grouped data; Spearman's rank correlation coefficient: Definition, examples of with and without repetition. **Regression**: Concept of dependent (response) and independent (predictor) variables, concept of regression, Types and prediction, difference between correlation and regression, Relation between correlation and regression. Linear Regression - Definition, examples using least square method and regression coefficient, coefficient of determination, properties.	**15**

References:

1. Fundamentals of Statistics, Vol. 1,Goon, A. M., Gupta, M. K. and Dasgupta, B., The World Press Pvt. Ltd., Calcutta, 6th Revised Edition, 1983
2. Fundamentals of Mathematical Statistics, Gupta, S.C. and Kapoor, V.K., S. Chand and Sons, New Delhi., 1987
3. Statistics Made it Simple: Do it yourself on PC, Sarma, K. V. S., Prentice Hall of India, New Delhi, 2001

Additional References:

1. Programmed Statistics, Agarwal, B. L., New Age International Publishers, New Delhi,2nd Edition, 2003
2. Statistics Using R, Purohit, S. G., Gore S. D., Deshmukh S. R., Narosa Publishing House, New Delhi, 2008
3. Schaum's Outline Of Theory And Problems Of Beginning Statistics, Larry J. Stephens, Mcgraw-Hill

Evaluation for Theory Courses (2 Credit Courses)

I. Continuous Internal Evaluation (CIA) - 20 Marks (40%)	**II. Semester End Theory Examination (SEE)- 30 Marks (60%)**
(i) **Mid-Term Class Test – 10 Marks** (ii) **Assignment/ Case study/ Presentations – 10 Marks**	**A Semester End Theory Examination of 1 hour duration for 30 Marks as per the paper pattern given below.**

Format of Theory question paper pattern

Question	**Based on**	**Options**	**Marks**
Q.1	Module I	Any 2 out of 4	10
Q.2	Module II	Any 2 out of 4	10
Q.3	Module I & II	Any 2 out of 4	10
		Total	30

- All questions shall be compulsory with internal choice within the questions.
- Each Question may be subdivided into sub questions as a, b, c, d, etc. & Each sub question carries 5 Marks.

Course/ Paper Title	**Web Technologies**
Course offered as	**VSC**
Course Code	**RUCSVSC206**
Semester	**II**
No. of Credits	**2 (60 hours)**
No. of lecture Hours/week	**4**

Sr No.	Course Learning Objectives:
CLO1	Implement the fundamental elements and attributes of web pages using HTML.
CLO2	Create static web pages by implementing HTML and CSS techniques.
CLO3	Design and implement a complete dynamic website, integrating XML.
CLO4	Examine and interpret the structure of dynamic web pages developed using JavaScript.
CLO5	Develop dynamic web pages using JavaScript and jQuery.

Course Outcome

	On completing the course, the student will be able to:
CO1	Develop web pages using HTML.
CO2	Implement dynamic web pages using HTML and CSS.
CO3	Develop a Program using XML.
CO4	Design and implement dynamic websites with a good aesthetic sense using JavaScript.
CO5	Design web pages using JavaScript and jQuery.

	Linux Programming - Practical
Sr No.	**Experiments**
1	Write a HTML program to demonstrate use of Text Formatting Tags like Heading-Paragraphs, Line Breaks, Background color, Text**:** Bold, Italic, Teletype, Underline, Strikethrough.
2	Write a HTML program for the demonstration of Lists. a. Unordered List b. Ordered List c. Definition List d. Nested List
3	Design a webpage that makes use of Table tags & Image tags.
4	Write a HTML program for demonstrating Hyperlinks. a. Navigation from one page to another. b. Navigation within the page.
5	Design a resume building form using html controls.
7	Write HTML for demonstration of cascading style sheets. a. Embedded stylesheets. b. External stylesheets. c. Inline styles

8	Design a webpage that make use of Cascading Style Sheets with a. CSS properties to change the background of a Page b. CSS properties to change Fonts and Text Styles c. CSS properties for positioning an element.
9	Write a JavaScript program to validate the following fields of the Registration page: First Name , Password , E-mail id, Mobile Number, Last Name and Address.
10	Develop and demonstrate JavaScript with POP-UP boxes and functions for the following problems: a) Input: Click on Display Date button using onclick( ) function Output: Display date in the textbox b) Input: A number n obtained using prompt Output: A multiplication table of numbers from 1 to 10 of n using alert c) Input: A number n obtained using prompt and add another number using confirm Output: Sum of the entire n numbers using alert
11	Write JavaScript code for a. Demonstrating different JavaScript Objects such as String, RegExp, Math, Date b. Demonstrating different JavaScript Objects such as Window, Navigator, History, Location, Document, c. Storing and Retrieving Cookies
12	Write a JavaScript that extracts the digits from the input and display the digits separated from one another by three spaces each. Example: Input 42339 output would be 4 2 3 3 9.
13	Write a JavaScript program that takes a number from a text field in the range of 0 to 999 and shows it in words. It should not accept four and above digits, alphabets and special characters.
14	Write a program for implementing XML documents for Student details.
15	Create a XML file with Internal / External DTD and display it using CSS & XSL
16	Design a webpage to handle asynchronous requests using AJAX on a. Mouseover b. button click
17	Write a program to load images from the server using AJAX.
18	Design a webpage with some jQuery animation effects.Also demonstrate how to limit character input in textarea including count in jQuery.
19	Design student registration form and display the data in another page using PHP
20	Write PHP scripts for performing CRUD operations using MYSQL.

Practical Evaluation (50 Marks)

Sr. No.	Practical Assessment	Marks
1	Practical Question 01	20
2	Practical Question 02	20
3	Viva	05
4	Journal	05

- Duration will be 02 hrs. for each practical course
- Certified Journal is compulsory for appearing at the time of practical Exam.
- Minimum 80% practical from each core subject are required to be completed.

Course/ Paper Title	**Application Development using python**
Course offered as	**SEC**
Course Code	**RUCSSECP207**
Semester	**II**
No. of Credits	**2** (60 hours)
No. of lecture Hours/week	**4**

Sr No.	Course Learning Objectives:
CLO1	Understand and explain the principles of object-oriented programming (OOP) in Python, including inheritance and polymorphism.
CLO2	Apply regular expressions and threading techniques to solve specific programming challenges and improve program efficiency.
CLO3	Demonstrate the ability to create, read, update, and delete files in Python, effectively managing file operations.
CLO4	Evaluate and implement exception handling strategies in Python applications to enhance robustness and error management.
CLO5	Design and develop applications that integrate database management, graphical user interfaces (GUIs), and networking capabilities using Python.
CLO6	Understand and explain the principles of object-oriented programming (OOP) in Python, including inheritance and polymorphism.

Course Outcome

	On completing the course, the student will be able to:
CO1	Apply OOP concepts in Python, including inheritance and polymorphism, to develop effective solutions.
CO2	Utilize regular expressions and threading to create efficient programs.
CO3	Analyze and manipulate files using Python to perform various operations.
CO4	Implement exception handling in Python applications to manage errors effectively.
CO5	Demonstrate knowledge of database interactions, GUI design, and networking concepts within Python.

	Python Programming - Practical
Sr No.	**Experiments**
1	A. Write a program to create a class Student having two data members & member functions. B. Write a program to create a class Employee having three data members & member functions for multiple objects.

2	Write a program to Python program to implement Inheritance. A. Single level B. Multi-Level C. Multiple D. Hierarchical
3	Write a Python program to implement the concept of Polymorphism A. Operator Overloading B. Method Overloading C. Method Overriding
4	Write a Python program to implement the types of constructors.
5	Write a Python program to implement the types of destructors.
6	Write a Python program to implement Abstract methods and classes.
7	Write a Python program to implement the concept of Interfaces.
8	Write a Python program to implement the concept of Interfaces.
9	Write a program to implement various file operations
10	Write a program to demonstrate the use of OS Module in python.
11	Write a Python program to demonstrate use of regular expressions for suitable application.
12	Write a Python Program to work with databases in Python to a. create & Connect to database. b. Creating and dropping tables
13	Write a Python Program to work with databases in Python to Insert and update values into tables.
14	Write a program to create server-client and exchange basic information.
15	Write a program to send an email and read the contents of the URL
16	Write a GUI Program to draw various shapes such as Line, Oval, circle etc
17	Write a GUI Program in Python to design an application that uses Different fonts and colors.
18	Write a GUI Program in Python to use various GUI Widgets and Layouts.
19	Design a GUI Registration form that contains all the widgets.
20	Write a program to Use Scrapy / Selenium/Beautiful soap for web mining.

Practical Evaluation (50 Marks)

Sr. No.	Practical Assessment	Marks
1	Practical Question 01	20
2	Practical Question 02	20
3	Viva	05
4	Journal	05

- Duration will be 02 hrs. for each practical course
- Certified Journal is compulsory for appearing at the time of practical Exam.
- Minimum 80% practical from each core subject are required to be completed.

Course/ Paper Title	**Business Communication**
Course offered as	**AEC**
Course Code	**RUCSAEC208**
Semester	**II**
No. of Credits	**2 (30 Hours)**
No. of lecture Hours/week	**2**

Sr No.	**Course Learning Objectives:**
CLO1	Explain the principles of effective communication.
CLO2	Use business communication techniques and new media tools.
CLO3	Assess the effectiveness of communication methods.
CLO4	Presentations and business correspondence.
CLO5	Produce professional documents and presentations.

Course Outcome

	On completing the course, the student will be able to:
CO1	Develop effective ways of business communication.
CO2	Practice professional presentation.
CO3	Demonstrate and understand use of new media in business communication.
CO4	Demonstrate business correspondence.
CO5	Develop Professional documents and presentations.

Detailed Syllabus

Module	**Title with content**	**No. of lectures**
I	**Business communication and Understanding its needs :** **Introduction:** Facing Today's Communication Challenges. Email Communication, importance of business communication in an organization,The Six Thinking Hat Method. **Effective Listening**: Business Presentations and Public Speaking, Conversations, Interviews**.** **Ethics in Business Communication:** Theories of Ethics, Correlation, between Values and Behavior, Nurturing Ethics, Importance of Work Ethics, Problems in the Absence of Work Ethics. **Academic and Professional Skills:** Professional Presentation: Nature of Oral Presentation, planning a Presentation, Preparing the Presentation, Delivering the Presentation	**15**

II	**Business Correspondence** **Theory of Business Letter** Writing Principles of Effective Letter Writing - 'You' Attitude, Jargon, Four C's of Communication – Correctness, Completeness, Conciseness, Courtesy Parts of a Business Letter Full Block Layout of a Business Letter Principles of Effective E-mail Writing **Personnel Correspondence** Statement of Purpose Letter of Recommendation Job Application Letter and Résumé Letter of Appointment Letter of Acceptance of Job Offer Letter of Appreciation Letter of Resignation **New Media in Communication** Impact of Technology Enabled Communication Types: Internet, Blogs, E-mail, Moodle, Social Media - Facebook, Twitter and What's App -Advantages & Disadvantages	**15**

References:

1. Business Communication 2nd Edition by Meenakshi Raman , Prakash Singh ,Oxford University
2.Strategies for improving your business communication by Prof. M. S. Rao Prakash Singh, Shroff Publishers and distributors

Additional References:

1.Communication Skills. Pune: Technical Publications, by Mathew, S.
2.Effective Business Communication. 7th ed. New Delhi: Tata McGraw Hill Publishing Company Limited by Murphy, H., Hildebrandt, H., and Thomas J.

Evaluation Pattern (2 Credit Courses) (50 marks)

- **Class Test (25 marks)**
- **Project/Assignment/Case Study (20 Marks)**
- **Attendance (05 marks)**

Course/ Paper Title	**Green Technologies**
Course offered as	**VEC**
Course Code	**RUCSVEC209**
Semester	**II**
No. of Credits	**2 (30 hours)**
No. of lecture Hours/week	**2**

Sr No.	**Course Learning Objectives:**
CLO1	Introduce the students to the core concepts of Green IT and the environment.
CLO2	Discuss about the life cycle of green devices and hardware.
CLO3	Describe energy-saving software techniques, data centres, data storage, networks and communications.
CLO4	Explain about assessing an organization's readiness for Green IT and Sustainable Software Development.
CLO5	Implement a strategic framework for Sustainable IT Services (SITS), Green Compliance and Emerging Carbon Issues.

Course Outcome

	On completing the course, the student will be able to:
CO1	Describe the environmental impacts of IT, Software, Data Centres.
CO2	Analyze the life cycle of IT devices, Data Storage and Networks and Communications.
CO3	Develop an enterprise IT Readiness, evaluate sustainable software development practices and Sustainable IT Services.
CO4	Explain about Protocols, ISO 14000-2004 Standards, Audits and Emerging Carbon Issues.

Detailed Syllabus

Module	Title with content	No. of lectures
I	**Green IT Overview:** Introduction, Environmental Concerns and Sustainable Development, Environmental Impacts of IT, Green I , Holistic Approach to Greening IT, Greening IT, Applying IT for Enhancing Environmental Sustainability, Green IT Standards and Eco-Labelling of IT , Enterprise Green IT Strategy, Green Washing, Green IT: Burden or Opportunity? **Green Devices and Hardware:** Introduction , Life Cycle of a Device or Hardware, Reuse, Recycle and Dispose. **Green Software:** Introduction , Processor Power States, Energy-Saving Software Techniques, Evaluating and Measuring Software Impact to Platform Power. **Green Data Centres:** Data Centres and Associated Energy Challenges, Data Centre IT Infrastructure, Data Centre Facility Infrastructure: Implications for Energy Efficiency, IT Infrastructure Management, Green Data Centre Metrics. **Green Data Storage:** Introduction,Storage Media Power Characteristics, Energy Management Techniques for Hard Disks, System-Level Energy Management. **Green Networks and Communications:** Introduction, Objectives of Green Network Protocols, Green Network Protocols and Standards.	**15**
II	**Sustainable Software Development:** Introduction, Current Practices, Sustainable Software, Software Sustainability Attributes, Software Sustainability Metrics, Sustainable Software Methodology, Defining Actions. **Enterprise Green IT Readiness:** Introduction, Readiness and Capability, Development of the G-Readiness Framework, Measuring an Organization's G-Readiness. **Sustainable IT Services: Creating a Framework for Service Innovation:** Introduction, Factors Driving the Development of Sustainable IT, Sustainable IT Services (SITS), SITS Strategic Framework. **Green Compliance: Protocols, Standards, and Audits:** Protocols and Standards, ISO 14000-2004 Standard, Various initiatives by stakeholders, Green Audits and types, Audit and use of Carbon emission management software. **Emerging Carbon Issues:** Technologies and Future: Future Carbon Landscape, Green ICT and Technology Trends, Cloud Computing, Nanotechnology, Quantum computing, Renewable energies, eco-design, Collaborative environmental intelligence.	**15**

References:

1. Harnessing Green IT: Principles and Practices, San Murugesan, G. R. Ganadharan, Wiley & IEEE
2. Green IT Strategies and Applications Using Environmental Intelligence, Bhuvan Unhelkar, CRC Press, 2016
3. Green Information and Communication Systems for a Sustainable Future, Rajshree Srivastava, Sandeep Kautish, Rajeev Tiwari. CRC Press, 2020

Additional References:

1. Emerging Green Technologies, Matthew N. O. Sadiku, Taylor and Francis (CRC Press), 2022
2. Sustainability Awareness and Green Information Technologies, Tomayess Issa, Springer, 2021
3. Environmental Sustainability Role of Green Technologies, P. Thangavel, and G. Sridevi, Springer, 2016
4. Green IT, Deepak Shikarpur, Vishwkarma Publications, 2014
5. Green Communications: Principles, Concepts and Practice- Samdanis et al, J. Wiley

Evaluation Pattern (2 Credit Courses) (50 marks)

- **Class Test (25 marks)**
- **Project/Assignment/Case Study (20 Marks)**
- **Attendance (05 marks)**

Name	Affiliation	Sign
Ms. Ritika Lala	Royal College of Arts, Science and Commerce Mira Road, Thane.	
Prof. Vipul Saluja	RD & SH National & SWA Science College Bandra (W), Mumbai.	
Dr. Anita Chaware	SNDT Women's University, Santacruz (W), Mumbai.	
Prof. Ashish Trivedi	Thakur College of Science and Commerce, Kandivali (E), Mumbai.	
Mr. Azhar Shaikh	Peepal Design , Pune	
Ms. Kamlesh Pal	Royal College of Arts, Science and Commerce Mira Road, Thane.	
Ms. Pramitha Santhumayor	Royal College of Arts, Science and Commerce Mira Road, Thane.	
Mr. Hasan Phudinawala	Royal College of Arts, Science and Commerce Mira Road, Thane.	
Mr. Vinay Dubey	Royal College of Arts, Science and Commerce Mira Road, Thane.	
Ms. Simran Hussain	Royal College of Arts, Science and Commerce Mira Road, Thane.	
Ms. Nitya Sinha	Royal College of Arts, Science and Commerce Mira Road, Thane.	
Ms. Anushka Padhye	Royal College of Arts, Science and Commerce Mira Road, Thane.	

ROYAL COLLEGE OF ARTS SCIENCE & COMMERCE DIST. THANE

Principal
ROYAL COLLEGE OF ARTS
SCIENCE & COMMERCE
Penkar Pada, Mira Road,
Dist. Thane, Pin- 401107.

Scanned with CamScanner