Education to Excel

SBRR MAHAJANA FIRST GRADE COLLEGE (Autonomous)

Jayalakshmipuram, Mysuru – 570 012 Karnataka, INDIA

Affiliated to University of Mysuru

Re-accredited by NAAC with 'A' Grade, College with Potential for Excellence

UG ☑ **PG** ☐

Syllabi of I and II Semester

BCA

Choice Based Credit System - 2019

BCA Programme (CBCS): 2019-20 onwards (Autonomous)

Semester	Core DSC	Credits	Elective DSE	Credits	SEC	Credits	AECC	Credits	Total Credits
I.	DSC-1 DSC-2 DSC-3	6 6 6					Kan/MIL-1 Eng – 1 EVS/COI	3 3 3	27
II.	DSC-4 DSC-5 DSC-6	6 6 6					Kan/MIL-2 Eng – 2 EVS/COI	3 3 3	27
III.	DSC-7 DSC-8 DSC-9	6 6 6					Kan/MIL-3 Eng – 3	3 3	24
IV.	DSC-10 DSC-11 DSC-12	6 6 6					Kan/MIL-4 Eng – 4	3 3	24
V.			DSE-1 DSE-2 DSE-3	6 6 6	SEC-1 SEC-2	2 2			22
VI.			DSE-4 DSE-5 DSE-6 (Major project)	6 6 6	SEC-3 SEC-4	2 2			22
TOAL		72		36		08		30	146

BCA Programme (CBCS): 2019-20 onwards (Autonomous)

Year	Semester	Course	Title	Hours / Week		Credits	Maximum Marks			Exam Duration	Total
							IA		Exam		
				L	T/ P	L+T+P	C1	C2	C3		Marks
I Year	I Semester	DSC 1	Programming Principles and C	4	0	4+0+0	10	10	80	3 Hours	150
			Programming Principles and C lab	0	4	0+0+2	05	05	40	3 Hours	
		DSC 2	Digital Electronics and Computer Organization	4	0	4+0+0	10	10	80	3 Hours	150
			Digital Electronics and Computer Organization Lab	0	4	0+0+2	05	05	40	3 Hours	
		DSC 3	Fundamentals of Information Technology	4	2	4+2+0	10	10	80	3 Hours	100
	II Semester	DSC 4	Object Oriented Programming with C++	4	0	4+0+0	10	10	80	3 Hours	150
			Object Oriented Programming with C++ lab	0	4	0+0+2	05	05	40	3 Hours	
		DSC 5	Operating Systems	4	0	4+0+0	10	10	80	3 Hours	150
			Operating Systems Lab	0	4	0+0+2	05	05	40	3 Hours	
		DSC 6	Discrete Mathematics	4	2	4+2+0	10	10	80	3 Hours	100

Discipline Specific Courses:

DSC-1: Programming Principles and C	4:0:2
DSC-2: Digital Electronics and Computer Organization	4:0:2
DSC-3: Fundamentals of Information Technology	4:2:0
DSC-4: Object Oriented Programming with C++	4:0:2
DSC-5: Operating SysteMS	4:0:2
DSC-6: Discrete Mathematics	4:2:0
DSC-7: Data Structures and ADA	4:0:2
DSC-8: Data Communication and Computer Networks	4:2:0
DSC-9: Database Management Systems	4:0:2
DSC-10: Object Oriented Programming with Java	4:0:2
DSC-11: Computer Graphics and Animation	4:0:2
DSC-12: Operation Research	4:2:0

List of Electives for both V and VI semesters (DSE)

(Select three of the following electives in V semester and two of the Elective in VI semester. (Without repetition)

Electives for V Sem

Software Engineering	(L:T:P::4:2:0)
.NET Programming	(L:T:P::4:0:2)
Python Programming	(L:T:P::4:0:2)
Web Applications	(L:T:P::4:0:2)
Digital Image Processing	(L:T:P::4:0:2)

Electives for VI Sem

Network Security	(L:T:P::4:2:0)
Cloud Computing and Big Data Analytics	(L:T:P::4:1:1)
Numerical and Statistical Analysis	(L:T:P::4:0:2)
Machine Learning	(L:T:P::4:0:2)

Skill Oriented Course (SEC)

Select two of the electives in V and VI semester.

SEC-1: DTP (Page Maker and CorelDraw)	(L:T:P::1:0:1)
SEC-2: Cyber Security	(L:T:P::1:0:1)
SEC-3: Software Testing	(L:T:P::1:0:1)
SEC-4: Android Programming	(L:T:P::1:0:1)

I Sem BCA

DSC-1: Programming Principles and C (LTP::4:0:2) 6 Credits

Course duration: 16 weeks with 4 hours of instruction per week.

Course Objective:
To develop understanding on programming concepts – logic, functional, and also enable learning problem solving skills and implementation of solution

Course outcome:
To apply programming knowledge/skills to design solutions to real world problems, including specifying, designing, implementing and validating solutions for new problems

UNIT I: Programming Concepts and Introduction to C language:
Program development life cycle: Problem definition, analysis, Design, Coding, Testing and debugging, Documentation and maintenance. Algorithm- Features, simple examples. Flowchart –Symbols used in a flowchart with suitable examples.
Overview of C: Importance of C, Basic structure of C program, executing a C program, sample C program. Constants ,variables- variables declaration and data types. C-character set, tokens, identifiers, assigning values to variables. Data type conversion.
Operators in C: Arithmetic operators, Relational Operators, Logical operators, assignment operators, increment and decrement operators, conditional operators, bitwise operators and special operators. Precedence of arithmetical expression, relational expression and logical expressions.
Input and output operations:
Input and output statements, reading a character: getchar(), writing a character: putchar(), formatted and unformatted i/o statements.

UNIT II: Control structures, Arrays and Strings.
Control structures:
Branching: if, if-else, nested if, else-if ladder, switch.
Looping: while, do-while and for loop. Jump statements and nested loops.
Arrays and Strings:
Arrays: Introduction, single dimensional array, two-dimensional arrays, initializing one and two- dimensional arrays, multi-dimensional arrays. Operations on arrays: traversal, insertion and deletion.
Strings: Declaring and initializing string variables, reading string from terminal, writing string to screen, putting strings together. Comparison of two strings, length of a string, copying a string, string operations using library functions & User defined functions.

UNIT III: Functions, Structures and Unions
Functions: Introduction, types of functions, need for user-defined functions, function call, types of arguments, nesting of functions, a multi function program, recursion, storage classes.

Structures: Definition and declaration of a structure, assigning and accessing the members of a structure, structure initialization, structure elements in memory, comparison of structure variables, structure with in the structure, array within structures.
Unions: Definition and declaration, accessing the members of a union. Comparison of structure with union.

UNIT IV: Pointers, Files and Graphics
Pointers: Advantages of pointers, declaration of pointer variable, pointer expressions, pointers and functions: call by value and call by reference, pointers and arrays, array of pointers, pointer to pointer.
Files: Definition, types of files. Creating Reference file. Modes of opening a file, formatted and unformatted i/o operations, random files.
Graphics: Introduction to graphics, graphics functions, drawing and filling images.

Reference Books:
1. E. Balagurusamy : Programming in ANSI C" Tata Mc Graw-Hill
2. Problem Solving with C -PHI(EEE). By - M.T.Somashekara.
3. S. ByronGottfried. : "Programming with C", Tata McGraw-Hill.
4. Yashawant Kanetkar : "Let us C"
5. Brain Verminghan & Dennis M. Ritchie "ANSI C Programming" (PHI)

I Sem BCA

DSC-2: Digital Electronics and Computer Organization (LTP::4:0:2) 6 Credits

Course duration: 16 weeks with 4 hours of instruction per week.

Course objectives:

- To introduce the basics involved in data representation, logic elements and design of digital logic circuits
- To teach basic organization of a computer and its principal components, viz, ALU, Control, Memory and Input/output.
- To enable the student to understand the design components of a digital subsystem that required realizing various components such as ALU, Control, etc.
- To expose students to the basic architecture of processing, memory and i/o organization in a computer system.

Course Outcomes:

- The student will be able to:
- Identify, understand and apply different number systems and codes.
- Understand the digital representation of data in a computer system.
- Understand the general concepts in digital logic design, including logic elements, and their use in combinational and sequential logic circuit design.
- Understand computer arithmetic formulate and solve problems, understand the performance requirements of systems.

Unit-I

Number Systems – Introduction- Decimal, Binary, Octal and Hexadecimal. Inter-Conversions, Addition, Subtraction, Multiplication and Division of Binary Number System. 1's and 2's Complement method in Binary Number System. Subtraction using 1's and 2's Compliment, Weighted Number System: Binary Coded Decimal (BCD), Addition of BCD Numbers. Non-Weighted Number System: Excess-3, Gray code Conversions, Gray and Binary Codes, Applications. Boolean Algebra: Basic laws, DeMorgan's theorem, Duality theorem, Sum Of Product method and Products of Sum method. Karnaugh map (Upto 4 Variables, Don't Care Condition).

Unit-II

Fundamentals of Logic Gates: Basic gates, Derived gates and Universal gates (Design).
Combinational and Sequential logic circuits - Half adder, Full adder, Half -Subtractor and Full-Subtractor. Flip-Flops - SR, D, JK, JK Master Slave, T Flip-flops, Decoders - 3 to 8 lines, Encoders-Octal to Binary. Multiplexer- 4 to 1 line, Counters-3 bits Binary Ripple counter, 3 bits synchronous binary counter.

Unit-III

Basic Organization of Computers, Von-Neumann M/c. CPU Organization: Fundamental Concepts: Fetching and storing a word in Memory, Register Transfer, Performing an Arithmetic & Logic Operation, Branching.

Unit-IV

Input / Output Organization: Peripheral Devices, Input – output Interface, I/O Bus, Synchronous Data Transfer and Asynchronous Data Transfer, sequential and Parallel computing, Modes of Transfer: Programmed I/O, Interrupt Driven I/O, Direct Memory Access (DMA), DMA Controller, I/O Channel & Processor, Interrupts. Memory Organization: Computers Memory System Overview, Characteristics of Memory System, Semi-Conductor, Organization, Memory cell Operation.

Reference Books:

1. Digital fundamentals-Thomas.D.Floyd. Malvino Leach, digital principles and application
2. Computer System Architecture Morris Mano PHI.
3. Computer Organization – by V.Carl Hamacher, Z.G.Vranesic, and S.G.Zaky, McGraw Hill.
4. Computer Organization & Design by – D.A.Patterson & J.L.Hennessy – Morgan Kaufmann Publishers (Elseviers)
5. Operating System by Madnick and Donovan.

I Sem BCA

DSC-3: Fundamentals of Information Technology (LTP:: 4:2:0) 6 Credits

Course duration: 16 weeks with 4 hours of instruction and 2 hours of tutorials per week.

Course Objective:

- To make students learn the basics of computer system and for its effective use in day to day life.

Course Outcomes:

- Be able to apply knowledge of computing device to analyze a problem in an application area, and define computing solution
- Be able to design, implement and evaluate a computer-based system to meet user needs
- Be able to effectively integrate IT based system solutions into the user environment

Unit - I

Introduction- Characteristics of Computers, Evolution of computers, Capabilities and limitations of computers, Generations of computers, Types of computers (micro, mini, main frame, super computers, Laptop, Tablets), Analog, Digital and hybrid computers, Block diagram of computer, computer applications - business and scientific. Data Organization: Drives, Files, Directories. Basic components of computers: Input devices- Keyboard, Mouse, Touch Screens, Joystick, Electronic Pen, Trackball, Scanning Devices-Optical Scanners, OCR, OMR, Bar Code Readers, MICR, Digitizer, Electronic card Reader, Image Capturing Devices-Digital Cameras.

Unit - II

Output Devices – Monitors - CRT, LCD/TFT, Printers- Dot matrix, Inkjet and Laser, Plotters-Drum, Flatbed, Screen Image Projector. Types of Memory (Primary And Secondary) RAM, ROM - PROM, EPROM, EEPROM, Secondary Storage Devices - Magnetic Tape, Magnetic Disks-Internal Hard Disk, External Hard Disk, Floppy Disks, Optical Disks-CD, CD-R, CD-RW, DVD, Solid State Storage-Flash Memory, USB Drives.

Unit- III

Computer Software- Software and its needs, Types of Software - System Software, Application Software, System Software - Operating System, Utility program, Programming languages - types, Assemblers, Compilers and Interpreter, Introduction to Operating System for PCs-DOS, Windows, Linux, Types of Programming Languages: Machine Level Language, Assembly Level Language, High Level Language, Virus - working, feature, types of viruses, virus detection, prevention and cure.

Unit - IV

Computer Security :The need for Computer Security, Basic Security Concepts, Threats to Users, Threats to Hardware, Threats to Data, Taking protective measures– Protecting Yourself, Protecting your privacy, Keeping your Data Secure. Introduction to GUI, Internet & www, email, browsers, search engines, internet chat, creating static web pages, E-Governance

- Introduction and Applications - Commerce basics, types of E-commerce, Benefits and limitations of E-commerce, EDI, Cyber law, Cyber banking, E-payment, Security, Cyber act, Legal and ethical issues in ecommerce, Cybercrime.

Reference Books:

1. Computer Fundamentals, V Rajaraman.
2. Computer Fundamentals, P.K Sinha
3. Computers Today, Mc Graw Hill publication.

Practical List

COMPUTER CONCEPTS AND C PROGRAMMING LAB

PART A

1. C programs to demonstrate switch statement.
2 .C programs to demonstrate Branching statements
3. C programs to demonstrate Looping statements.
4. C programs to demonstrate one dimensional Array.
5. C programs to demonstrate two dimensional Array.

PART B

1. C Programs to perform String operation using library functions
3. C programs to demonstrate functions.
4. C programs to demonstrate recursive function.
6. C programs to demonstrate structure.
7. C programs to demonstrate function and pointers
8. C Programs to demonstrate files.
9. C programs to demonstrate geometrical shapes and filling colors using graphics.

Practical List

DIGITAL ELECTRONICS AND COMPUTER ORGANIZATION LAB

PART A - DIGITAL ELECTRONICS LAB

1. Verification of basic gates [OR, AND, NOT] and EX-OR, EX-NOR.
2. Verification of other gates using only NAND gate and NOR gate [Universal gate].
3. Realization of Boolean expressions using basic gates.
4. Simplification of Boolean expressions using Karnaugh map method.
5. Construction of Half and Full Adders.
6. Construction of BCD to decimal decoder.
7. Implementation of SR flipflop and clocked SR flipflop.
8. Implementation of JK Flip-flop.

PART B - FIT LAB

MS Word Exercise

1. Decorate word document with page border, content border, add pattern, insert image and write beautiful text in the document.
2. Design cover letter and also demonstrate Mail Merge feature in Microsoft Word.
3. Create letterhead of any company or institution and insert the watermark with that company name in the document.
4. Create bill/leaflets/brochures/Business Cards/Certificate/Party Invitations/Snapshot Calendar using text, various shapes and colors.
5. Design advertisement in Microsoft Word.
6. Create organization charts.
7. Make books content page or index page.
8. Practice hyperlink and create links between word document texts to Play songs from Microsoft word text, create the link between internal and external files.
9. Design or create double column book or newspaper in the word document.
10. Demonstrate OLE feature in MS Word.
11. Using table feature enter appropriate data, sort records and apply formulae on table.

MS Excel Exercise

1. Creating a basic spreadsheet by entering text, numbers and creating formulas.
2. Formatting cells, columns, numbers, dates and worksheets and Data validation.
3. Use of Absolute references, Range names, Conditional formulae.
4. Using built in functions to create a spreadsheet to perform "what if?" calculations.
5. Demonstrate creating charts.
1. Demonstrate use of conditional formatting tool.
2. To sort data and print portions of a worksheet.
3. Basic statistics functions to analyse data.
4. To dress up a table using special formats and how to export a table or chart into a Microsoft Word document.
5. Demonstrates a basic cost-benefit analysis using Excel.
6. To consolidate several worksheets into one and to link several worksheets to a master worksheet.

7. Worksheet to calculate descriptive statistics (e.g., mean, standard deviation, frequency distribution, correlation).
8. Use of Pivot Table to work with multiple worksheets.
9. Demonstrate the use of VLOOKUP function.

MS Access Exercise

1. Creating a Table, Entering Table Data, creating a Form, Entering Data into a Form.
2. Creating Queries, Multi-table queries.
3. Creating Relationships between tables, enforcing referential integrity.
4. Grouping, Totals and Joins.

MS PowerPoint Exercise

1. Creating a Slide Presentation, slide footers, section headers, and formatted text.
2. Selecting themes and modifying themes.
3. Working with graphical and multimedia elements – audio, images, picture styles, charts, tables.
4. Applying transitions and animations.
5. Custom Slide Show, setting up PowerPoint Show.

II Sem BCA

DSC-4: Object Oriented Programming with C++
(LTP::4:0:2) 6 Credits

Course duration: 16 weeks with 4 hours of instruction per week.

Course Objectives:

- To introduce C++ syntax and semantics to write programs
- To enable understand the fundamentals of object-oriented programming using C++
- Create, Debug and test a software application using the C++ programming language.

Course Outcomes:

- On successful completion of this course the student should be able to:
- Create C++ programs that solve simple real-world problems.
- Validate user input, perform a test plan to validate a C++ program and document C++ program.

Unit-1

Introduction: Comparison of POP and OOP, Concepts of Object-Oriented Programming, Structure of C++ Program.

Fundamentals: Tokens, Keywords, Identifiers and Constants, Basic Data Types, User-defined data types, Derived data Types, Symbolic constants, Type compatibility, Declaration of variables, Dynamic initialization of variables, Reference variables, Operators in C++, Scope Resolution Operator, Member dereferencing operators, Memory Management operators, Manipulators, Type Cast Operator, Expressions and their types, Special assignment expressions, Implicit and Explicit conversions, Operator Overloading, Operator Precedence, Control Structures.

Unit-2

Functions: The main function, Function prototyping, Call by Reference, Return by Reference, Default arguments, Function overloading, Friend and Virtual functions.

Classes and Objects: Specifying a Class, Defining member functions, Making an Outside function Inline, Nesting of member functions, Private member functions, Arrays within a Class, Static data members, Static member functions, Arrays of Objects, Objects as function arguments, friendly functions, Returning Objects, Constant member functions, Pointers to members.

Unit-3

Constructors and Destructors: Constructors, Parameterized Constructors, Multiple Constructors in a class, Constructors with default arguments, Dynamic initialization of objects, Copy constructor, Dynamic Constructor and Destructors.

Operator Overloading and Type Conversions: Defining operator Overloading, Overloading Unary operators, Overloading Binary operators, Rules for overloading operators, Type conversions.

Unit-4

Inheritance: Defining a derived class, Single Inheritance, Protected members, Multilevel Inheritance, Multiple Inheritance, Hierarchical Inheritance, Hybrid Inheritance, Virtual Base classes, Abstract classes, Constructors in derived classes, Nesting of classes.

Pointers, virtual functions, polymorphism: Pointers to objects, This Pointer, Pointers to derived classes, Virtual functions, Pure virtual functions, Virtual Constructors and Destructors.

Reference Books:

1. E Balagurusamy, **Object Oriented Programming with C++**, Tata McGraw hill Publication.
2. D Ravichandran, **Programming with C++**, McGraw hill.
3. Robert Lafore, **Oriented Programming in C++**, Galgotia Publications Pvt. Ltd.
4. The C++ Programming Language, Bjarne Stroustrup

II Sem BCA

DSC-5: OPERATING SYSTEMS
(L:T:P::4:0:2) 6 Credits

Course duration: 16 weeks with 4 hours of instruction per week.

Course Objectives:

- To make students understand the purpose, role, structure, functions, application of Operating systems
- To understand services provided by Operating Systems

Course Outcomes:

After completion of the course a student should be able to:

- Analyze the structure of Operating Systems and basic architectural components involved in design
- Analyze the various resource management techniques
- Interpret the mechanisms adopted for file sharing
- conceptualize the components involved in designing a contemporary Operating Systems
- Be familiar with various types of Operating Systems

Unit 1: Introduction to Systems Software

Introduction, Systems software and Machine architecture, Assembler-Introduction, Design of Assembler, Loaders and its types, Linkers, Macros and text editors, Interpreters and Compilers.

Unit 2: Introduction and Process Management

Operating System– Introduction, Definition, Functions, Features and Types- Batch Systems, Multiprogramming, Time Sharing, Real Time, Parallel and Distributed Systems, Computing Environments – Traditional, Client- Server, Peer-to-Peer and Web based, Process Management: Process concept – meaning of process, Types – System and User, process State, Process State Diagram, Process Control Block, Threads, Scheduling Queues, schedulers, Context switch, CPU- I/O burst cycle, CPU bound/IO bound process, Dispatcher, Scheduling criteria.

Unit 3: Scheduling Algorithms and Deadlocks

Scheduling algorithms: Preemptive and Non-Preemptive, First-Come-First-Served (FCFS), Shortest Job First (SJF), Priority Scheduling, Round Robin. Real time scheduling with pre-emption and Non-preemption.

Deadlocks: Definition, System model, Deadlock characterization – Necessary Conditions, Resource Allocation Graph, Deadlock handling methods, Prevention, Avoidance and Detection, Recovery Schemes.

Unit 4: Memory Management

Introduction to Memory Management, Functions , Partitioned memory- Single partition, Multiple partition (MFT & MVT), Fragmentation, Memory Management Technique - Paging, Virtual Memory Concepts, Segmentation, Demand paging-Page fault, Page replacement algorithms - FIFO, LRU, Optimal page replacement, UNIX-Introduction and basics.

Reference Books:

1. System programming – John. J. Donovan.
2. System Software – Leland L. Beck, Addison Wesley.
3. Operating System Concepts, Abraham Silberschatz and Peter Baer Galvin, Addison wesley.
4. Operating System Concepts & Design, Milan Milonkovic, McGRaw Hill.
5. Operating SysteMS, Stallings, Pearson Edition.
6. Operating Systems Concepts, Tanenbaum, Pearson Education.
7.Operating System by Stuart E Madnick, John J. Donovan.

II Sem BCA

DSC-6 Discrete Mathematics (LTP:: 4:2:0) 6 Credits

Course duration: 16 weeks with 4 hours of instruction and 2 hours of tutorials per week.

Course Objectives

- To simplify and evaluate basic logic statements including compound statements, implications, inverses, converses, and contrapositives using truth tables and the properties of logic.
- To express a logic sentence in terms of predicates, quantifiers, and logical connectives
- To develop logical thinking and its application to computer science
- To develop ability to present a coherent and mathematically accurate argument.

Course Outcomes

After completing this course satisfactorily, a student will:

- Be able to construct simple mathematical proofs and possess the ability to verify them.
- Have substantial experience to comprehend formal logical arguments.
- Be skillful in expressing mathematical properties formally via the formal language of propositional logic and predicate logic .
- Be able to specify and manipulate basic mathematical objects such as sets, functions, and relations and will also be able to verify simple mathematical properties that these objects possess.
- Acquire ability to describe computer programs (e.g. recursive functions) in a formal mathematical manner.
- Be able to apply basic counting techniques to solve combinatorial problems.
- Gain experience in using various techniques of mathematical induction (weak, strong and structural induction) to prove simple mathematical properties of a variety of discrete structures.

UNIT –I

Set Theory: Sets and Subsets, Set Operations and the Laws of Set Theory, Counting and Venn Diagrams, Cartesian Products and Relations, Functions–One-to-One, Onto Functions, Function Composition and Inverse Functions; Properties of Relations, Computer Recognition–Zero-One Matrices and Directed Graphs, Partial Orders – Hasse Diagrams, Equivalence Relations and Partitions.

UNIT – II

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

UNIT –III

Mathematical Induction and Recursion: Sequences and summations, Mathematical Induction, The Well Ordering Principle, Recursive Definitions, Structural Induction, Recursive algorithms. Counting: Basics of counting, Pigeonhole Principle, Permutation and Combinations, Binomial coefficients. Graphs: Introduction, Representing Graphs & Graph

Isomorphism, Connectivity, Euler and Hamilton Paths, Shortest path problems, Planar Graphs, Graph colouring.

UNIT – IV

Trees: Introduction, Applications of Trees, Tree Traversal, Spanning Trees, Minimum Spanning Trees.

Introduction to Numerical Techniques:

Characteristics, Applications of Numerical Techniques, Process involved in Numerical Technique, Comparison of Analytical techniques with Numerical Techniques**,** consequences of normalized floating point representation of numbers, Errors and its significances. Equations and its types.

Reference Books:

1. Ralph P. Grimaldi, “Discrete and Combinatorial Mathematics”, Pearson Education.
2. Kenneth H. Rosen, “Discrete Mathematics and its Applications”, McGraw Hill.
3. Jayant Ganguly, “A Treatise on Discrete Mathematical Structures”, Sanguine Pearson.
4. D.S. Malik and M.K. Sen, “Discrete Mathematical Structures: Theory and Applications”, Thomson.
5. Thomas Koshy, “Discrete Mathematics with Applications”, Elsevier.
6. E. Balagurusamy, “Numerical Methods”, McGraw Hill.

Practical List

Object Oriented Programming with C++

PART-A

1. C++ programs to implement Basic Programs without objects
2. C++ programs to demonstrate functions.
3. C++ programs to demonstrate Branching statements.
4. C++ program to demonstrate Looping statements.
5. C++ program to demonstrate classes and objects.
6. C++ program to illustrate the use of static member function.
7. C++ program to illustrate the friend function..
8. C++ program to show the use of copy constructor.
9. C++ program to illustrate pointers to objects.
10. C++ program to demonstrate getline() and write() function.

PART - B

1. C++ program to accept two times (hh:mm:ss) to find subtraction of two times.
2. C++ program to implement Shopping mart using a class with arrays of objects.
3. C++ program to implement banking scheme.
4. C++ program to illustrate the use of over loaded constructor.
5. C++ program to construct variables at run time using dynamic initialization.
6. C++ program to find the largest value among the set of parameters using overloaded function.
7. C++ program to demonstrate operator overloading.
8. C++ program to implement Inheritance.
9. C++ program to illustrate the implementation of virtual base class.
10. C++ program to illustrate the use of array of pointer to objects.
11. C++ program to implement file operations.

Practical List

SYSTEM SOFTWARE AND OPERATING SYSTEMS LAB

PART A: UNIX LAB

1. Write a shell script to exchange the contents of two variables.
2. Write a shell script, which accepts three subject marks scored by a student and declare the result.
3. Write a shell script to print integer numbers from 1 to 20.
4. Write a shell script to perform arithmetic operation on two number depending on +, -, * and /.
5. Write an interactive shell script to display a menu and perform the following task:
    i. Renaming a file
    ii. Deleting a file
    iii. Copying a file
    iv. Exit
6. Write a shell script which counts the number of lines in a file.
7. Write a c program to
    a. Display the PID of parent and PID of child.
    b. Copy the contents of one file into the other using command line arguments.
8. Assume a file with following information
    FirstName, MiddleName, Age
    ---------- ---------- -----
    Write a shell script
    i. To Sort the first name in alphabetical order.
    ii. Sort the age in terms of ascending order.
    iii. Sort the age in terms of descending order.
    iv. Sort the middle name in alphabetical order.
9. Write shell script to find factorial of a given number

PART-B: OPERATING SYSTEM LAB

10. Write ‘C’ program to illustrate FCFS scheduling
11. Write ‘C’ program to illustrate SJF scheduling
12. Write ‘C’ program to illustrate Round robin scheduling
13. Write ‘C’ program to illustrate Priority scheduling

Scheme of Valuation for Practical

C1 and C2 are internal tests to be conducted during 8th and 16th weeks respectively of the semester. C3 is the semester-end examination conducted for 3 hours. The student will be evaluated on the basis of Analytical and Programming skills, comprehension and based on the final result.

The students have to compulsorily submit the practical record during C1 and C2. For C3, the record has to be certified by the Head of the Department.

- The student is evaluated for 20 marks in C1 and C2 as per the following scheme:

The marks scored are then normalized for 10.

- The student is evaluated for 40 marks in C3 as per the following scheme:

(The Examiners should set two Programs, one from Part-A and another from Part-B)

Heading	Marks
Program Writing	6*2=12
Implementation & Result	8*2=16
VIVA	08
Record	04
Total	**40**

Question Paper Pattern

DSC Courses:

Max Marks: 80 **Time: 3 hours**

PART A

Answer 10 out of 12 Questions. [3 Questions to be given from each unit] (10 X 2 = 20)

(Q1…to ...12)

PART B

Answer all questions. Answer any one sub question from each main question. (4×5 = 20)

[2 Questions to be given from each unit]

From	Q 13	a.	to	Q 16	a.
		b.			b.

PART C

Answer all questions. Answer any one sub question from each main question. (4x10 =40)

[2 Questions to be given from each unit and internal split is allowed]

From	Q 17	a.	to	Q 20	a.
		b.			b.