POORNAPRAJNA COLLEGE (AUTONOMOUS), UDUPI
NAAC Re-Accredited "A+" (3.27 CGPA)
(Promoted and Managed by Udupi Shree Adamaru Matha Education Council, Bengaluru)

SYLLABUS FOR UNDERGRADUATE PROGRAM (UG) OF COMPUTER APPLICATION CURRICULUM FRAMEWORK

COURSE PATTERN AND SCHEME OF EXAMINATION FOR UG COURSES

FRAMED ACCORDING TO THE STATE EDUCATION POLICY (SEP 2024)

BCA GENERAL

I & II SEMESTERS

TO IMPLEMENT FROM THE ACADEMIC YEAR 2025-26

BOARD OF STUDIES IN COMPUTER SCIENCE
POORNAPRAJNA COLLEGE (AUTONOMOUS), UDUPI - 576101

Preamble

Computer Application (CA) has been evolving as an important branch of science and technology in last two decade and it has carved out a space for itself like computer science and engineering. Computer application spans theory and more application and it requires thinking both in abstract terms and in concrete terms.

The ever – evolving discipline of computer application has strong connections to other disciplines. Many problems in science, engineering, health care, business, and other areas can be solved effectively with computers and its applications, but finding a solution requires both computer science expertise and knowledge of the particular application domain.

Computer science has a wide range of specialties. These include Computer Architecture, Software Systems, Graphics, Artificial Intelligence, Mathematical and Statistical Analysis, Data Science, Computational Science, and Software Engineering.

Universities and other HEIs introduced programmes of computer application. Information Technology is growing rapidly. Increasing applications of computers in almost all areas of human endeavour has led to vibrant industries with concurrent rapid change in technology. Unlike other basic disciplines, developing core competency in this discipline that can be reasonably stable becomes a challenge.

In India, it was initially introduced at the Master (postgraduate) level as MCA and M.Tech. Later on, engineering programmes such as B.Tech and B.E in Computer Science & Engineering and in Information Technology were introduced in various engineering College/Institutions to cater to the growing demand for trained engineering manpower in IT industries. Parallelly, BCA, BSc and MSc programmes with specialization in Computer Science were introduced to train manpower in this highly demanding area.

BCA aimed at undergraduate level training facilitating multiple career paths. Students so graduated, can take up postgraduate programmes in CS or MCA leading to research as well as R&D, can be employable at IT industries, or can pursue a teaching profession or can adopt a business management career.

BCA aims at laying a strong foundation of computer application at an early stage of the career. There are several employment opportunities and after successful completion of BCA, graduating students can fetch employment directly in companies as programmer, Web Developer, Software Engineer, Network Administrator, Data Scientist, or AI/ML personnel.

The Program outcomes in BCA are aimed at allowing flexibility and innovation in design and development of course content, in method of imparting training, in teaching learning process and in assessment procedures of the learning outcomes. The emphasis in BCA courses, in outcome-based curriculum framework, help students learn solving problems, accomplishing IT tasks, and expressing creativity, both individually and collaboratively. The proposed framework will help Students learn programming techniques and the syntax of one or more programming languages.

All students must, therefore, have access to a computer with a modern programming language installed. The computer science framework does not prescribe a specific language. The teacher and students will decide which modern programming languages students will learn. More importantly, students will learn to adapt to changes in programming languages and learn new languages as they are developed.

The present Curriculum Framework for BCA degrees are intended to facilitate the students to achieve the following.

- To develop an understanding and knowledge of the basic theory of Computer Science and Information Technology with good foundation on theory, systems and applications such as algorithms, data structures, data handling, data communication and computation.

- To develop the ability to use this knowledge to analyze new situations in the application domain.

- To acquire necessary and state-of-the-art skills to take up industry challenges. The objectives and outcomes are carefully designed to suit to the above-mentioned purpose.

- The ability to synthesize the acquired knowledge, understanding and experience for a better and improved comprehension of the real-life problems.

- To learn skills and tools like mathematics, statistics and electronics to find the solution, interpret the results and make predictions for the future developments.

- To formulate, to model, to design solutions, procedure and to use software tools to solve real world problems and evaluate.

The objectives of the Programme are:

- The primary objective of this program is to provide a foundation of computing principles and business practices for effectively using/managing information systems and enterprise software.

- It helps students analyze the requirements for system development and exposes students to business software and information systems.

- This course provides students with options to specialize in legacy application software, system software or mobile applications.

- To produce outstanding IT professionals who can apply the theoretical knowledge into practice in the real world and develop standalone live projects themselves.

- To provide opportunity for the study of modern methods of information processing and its applications.

- To develop among students the programming techniques and the problem-solving skills through programming.

- To prepare students who wish to go on to further studies in computer science and related subjects.

- To acquaint students to Work effectively with a range of current, standard, Office Productivity software applications.

Programme Outcomes: BCA (3 Years) Degree

- Discipline knowledge: Acquiring knowledge on basics of Computer Science and ability to apply to design principles in the development of solutions for problems of varying complexity.

- Problem Solving: Improved reasoning with strong mathematical ability to Identify, formulate and analyze problems related to computer science and exhibiting a sound knowledge on data structures and algorithms.

- Design and Development of Solutions: Ability to design and development of algorithmic solutions to real world problems and acquiring a minimum knowledge on statistics and optimization problems. Establishing excellent skills in applying various design strategies for solving complex problems.

- Programming a computer: Exhibiting strong skills required to program a computer for various issues and problems of day-to-day applications with thorough knowledge on programming languages of various levels.

- Application Systems Knowledge: Possessing a sound knowledge on computer application software and ability to design and develop app for applicative problems.

- Modern Tool Usage: Identify, select and use a modern scientific and IT tool or technique for modeling, prediction, data analysis and solving problems in the area of Computer Science and making them mobile based application software.

- Communication: Must have a reasonably good communication knowledge both in oral and writing. Project Management: Practicing of existing projects and becoming independent to launch own project by identifying a gap in solutions.

- Ethics on Profession, Environment and Society: Exhibiting professional ethics to maintain the integrality in a working environment and also have concern on societal impacts due to computer-based solutions for problems. Lifelong Learning: Should become an independent learner. So, learn to learn ability. Motivation to take up Higher Studies: Inspiration to continue educations towards advanced studies on Computer Science.

Suggested programme structure for the Under Graduate Programmes

[BCA, BCA (A.I & M.L), BCA (D.A)]

Semester	Course 1	Course 2	Course 3	Elective / Optional	Course	Language	Compulsory	Total Credit	Total Working hour
I	5 (3T+2P)	5 (3T+2P)	5 T			3+3	2	23	4+4+4+4+5+4+4+2=31
II	5 (3T+2P)	5 (3T+2P)	5T			3+3	2	23	4+4+4+4+5+4+4+2=31
III	5 (3T+2P)	5 (3T+2P)	5T	2		3+3		23	4+4+4+4+5+4+4+2=31
IV	5 (3T+2P)	5 (3T+2P)	5T	2		3+3	2	25	4+4+4+4+5+2+4+4+2=33
V	8[(2x3T)+2P]	8[(2x3T)+2P]	8[(2x3T)+2P]				2	26	3+3+4+3+3+4+3+3+4+2=32
VI	3T	3T	3T		3T	Project work 12		24	3+3+3+3+24=36
								144	

Note:

- Course1 and Course2: I to IV Semester: Theory 3 credit=4 contact hours & Practical 2 credit=4 contact hours
- Course3: I to IV Semester: Theory 5 credit=5 contact hours
- Course1, Course2 and Course3: V and VI Semester: Theory 3 credit=3 contact hours & Practical 2 credit=4 contact hours
- Elective/Optional: 2 credit=2 contact hours
- Languages: 3 credit=4 contact hours
- Compulsory: 2 credit=2 contact hours

CURRICULUM STRUCTURE FOR I TO II SEMETER BCA

Semester I								
Sl. No.	Course Code	Title of the Course	Category of Courses	Teaching Hours per Week	SE E	IA	Total Marks	Credits
1		Language-I	Lang	4	80	20	100	3
2		Language-II	Lang	4	80	20	100	3
3	BCCACS101	Fundamentals of Computers	Core	4	80	20	100	3
4	BCCACS102	Programming in C	Core	4	80	20	100	3
5	BCCACS103	Discrete Mathematics for Computer Applications	Core	5	80	20	100	5
6	BCCAPS101	Information Technology Lab	Practical	4	40	10	50	2
7	BCCAPS102	C Programming Lab	Practical	4	40	10	50	2
8		Constitution/Values	Compulsory	2	40	10	50	2
Sub - Total				31	520	130	650	23

Semester II								
Sl. No	Course Code	Title of the Course	Category of Courses	Teaching Hours per Week	SE E	IA	Total Marks	Credits
1		Language-I	Lang	4	80	20	100	3
2		Language-II	Lang	4	80	20	100	3
3	BCCACS201	Data Structures	Core	4	80	20	100	3
4	BCCACS202	Object Oriented Programming using Java	Core	4	80	20	100	3
5	BCCACS203	Computational Mathematics	Core	5	80	20	100	5
6	BCCAPS201	Data Structures Lab	practical	4	40	10	50	2
7	BCCAPS202	Object Oriented Programming Lab	practical	4	40	10	50	2
8		Constitution/Values	Compulsory	2	40	10	50	2
Sub - Total				31	520	130	650	23

CURRICULUM STRUCTURE FOR I SEMESTER BCA GENERAL

Semester I								
Sl. No.	**Course Code**	**Title of the Cour*se***	**Category of Courses**	**Teaching Hours per Week**	**SE E**	**IA**	**Total Marks**	**Credits**
1	**BCCACS101**	**Fundamentals of Computers**	**Core**	**4**	**80**	**20**	**100**	**3**
2	**BCCACS102**	**Programming in C**	**Core**	**4**	**80**	**20**	**100**	**3**
3	**BCCACS103**	**Discrete Mathematics for Computer Applications**	**Core**	**5**	**80**	**20**	**100**	**5**
4	**BCCAPS101**	**Information Technology Lab**	**Practical**	**4**	**40**	**10**	**50**	**2**
5	**BCCAPS102**	**C Programming Lab**	**Practical**	**4**	**40**	**10**	**50**	**2**

Program Name	**BCA**	Semester	**I**
Course Title	**Fundamentals of Computers (Theory)**		
Course Code:	**BCCACS101**	No. of Credits	**03**
Contact hours	**4 Hours per week**	Duration of SEA/Exam	**3 Hours**
Formative Assessment Marks	**20**	Summative Assessment Marks	**80**

Course Outcomes (COs):

After completing this course satisfactorily, a student will be able to:

- Understand the fundamentals of computer system
- Identify different components within the computer system
- Understand different types of input and output devices
- Demonstrate the working concepts of different devices connected to computer
- Explain different generations of programming languages and their significance
- Understand the use of Word processing, Spreadsheet, Presentation and DBMS applications
- Understand Digital computer and digital systems functioning

Topics	**Book & Chapter**	**Page No. / Section No.**
UNIT I		
Computer Basics: Introduction, Characteristics computers, Evolution computers, Generations of computers, Classification of computers, the computer system, Application of computers.	Book 1 Chapter 1	1.1 to 1.6
Computer Architecture: Introduction, Central processing unit – ALU, Registers, Control unit, system bus, main memory unit, cache memory.	Book 1 Chapter 2	2.1, 2.2
Input devices: Introduction, Types of input devices, Keyboard, Mouse, Track ball, Joystick light pen, Touch screen and track pad. Speech recognition, digital camera, webcam, flatbed scanner.	Book 1 Chapter 4	4.1, 4.2: 4.2.1, 4.2.2, 4.2.3, 4.2.4, 4.2.5, 4.2.6 (Excluding the working of devices)
Output devices: Types of output, Classification of output devices, Printers–Dot matrix, Ink-jet, Laser, Hydra, Plotter, Monitor – CRT, LCD, Differences between LCD and CRT	Book 1 Chapter 4	4.3: 4.3.1, 4.3.2, 4.3.4 (Excluding the working of devices and Daisy wheel Printer)

UNIT II		
Computer software: Introduction, software definition, relationship between software and hardware, software categories.	Book 1 Chapter 11	11.1, 11.2, 11.3
Computer programming languages: Introduction, developing a program, Program development cycle, Types of programming languages, generation of programming languages, Features of a good programming language.	Book 1 Chapter 10	10.1, 10.9, 10.10, 10.11
Problem Solving techniques: Introduction, Problem solving procedure.	Book 2 Chapter 1	1.1, 1.2
Algorithm: Steps involved in algorithm development, Algorithms for simple problems (To find largest of three numbers, factorial of a number, check for prime number, check for palindrome, Count number of odd, even and zeros in a list of integers)	Book 1 Chapter 10	10.2
Flowchart: Definition, advantages, Symbols used in flow charts. Flowcharts for simple problems mentioned in algorithms. Pseudocode.	Book 1 Chapter 10	10.3, 10.5
UNIT III		
Digital Computers and Digital System: Introduction to Number System, Decimal number, Binary number, Octal and Hexadecimal numbers, Number base conversion, Complements, Binary codes, Binary arithmetic, Addition, Subtraction in the 1's and 2's complements system, Subtraction in the 9's and 10's complement system.	Book 3 Chapter 1	1.2, 1.3, 1.4, 1.5, 1.6
Boolean Algebra: Basic definitions, Axiomatic definition of Boolean algebra, Basic theorems and properties of Boolean algebra, Venn diagram.	Book 3 Chapter 2	2.1, 2.2, 2.3
UNIT IV		
Digital logical gate: Boolean functions, Canonical and Standard forms, Minterms, Maxterms, other logic operations, Digital logic gates, Universal gates.	Book 3 Chapter 2 Chapter 4	2.4, 2.5, 2.6, 2.7 (Excluding 2.7.1), 4.7.1
Simplification of Boolean function: The map method, Two and three variable maps, Four variable maps, Don't care conditions, Product of sum simplification.	Book 3 Chapter 3	3.1, 3.2, 3.3, 3.5, 3.8

Text Books:

1. ITL Education Solution Limited, Introduction to Information Technology, Second Edition, Pearson
2. How to Solve it by Computer, R G Dromey, Prentice Hall
3. M. Morris Mano, Digital Logic and Computer design, PHI, 2015

References Books:

1. Pradeep K. Sinha and Priti Sinha, Computer Fundamentals, Sixth Edition, BPB Publication.
2. David Riley and Kenny Hunt, Computational thinking for modern solver, Chapman & Hall/CRC.
3. J. Glenn Brookshear, Computer Science: An Overview, Twelfth Edition, Addison-Wesley

Program Name	**BCA**	Semester	**I**
Course Title	**Programming in C (Theory)**		
Course Code:	**BCCACS102**	No. of Credits	**03**
Contact hours	**4 Hours per week**	Duration of SEA/Exam	**3 Hours**
Formative Assessment Marks	**20**	Summative Assessment Marks	**80**

Course Outcomes (COs):

After completing this course satisfactorily, a student will be able to:

- Confidently operate Desktop Computers to carry out computational tasks
- Understand working of Hardware and Software and the importance of operating systems
- Understand programming languages, number systems, peripheral devices, networking, multimedia and internet concepts
- Read, understand and trace the execution of programs written in C language
- Write the C code for a given problem
- Perform input and output operations using programs in C
- Write programs that perform operations on arrays

Topics	**Book**	**Chapter / Page No. / Section No.**
UNIT I		
Overview of C: History of C, Importance of C Program, Basic structure of a C-program, Execution of C Program.	Book 1	Chapter 1
C Programming Basic Concepts: Character set, C token, Keywords and identifiers, Constants, Variables, data types, Declaration of variables, assigning values to variables, defining symbolic constants.	Book 1	Chapter 2
Input and output with C: Formatted I/O functions - printf and scanf, control stings and escape sequences, output specifications with printf functions; Unformatted I/O functions to read and display single character and a string - getchar, putchar, gets and puts functions.	Book 1	Chapter 4

UNIT II		
Operators & Expressions: Arithmetic operators; Relational operators; Logical operators; Assignment operators; Increment & Decrement operators; Bitwise operators; Conditional operator; Special operators; Operator Precedence and Associatively; Evaluation of arithmetic expressions; Type conversion.	Book 1	Chapter 3
Control Structures: Decision Making and Branching - Decision making with if statement, simple if statement, the if else statement, nesting of if … else statements, the else if ladder, the switch statement, the?: operator, the go to statement.	Book 1	Chapter 5
Decision making and looping - The while statement, the do statement, for statement, nested loops, exit, break, jumps in loops.	Book 1	Chapter 6
UNIT III		
Derived data types in C: Arrays - declaration, initialization and access of one-dimensional and two-dimensional arrays. programs using one- and two-dimensional arrays, sorting and searching arrays.	Book 1	Chapter 7
Handling of Strings: Declaring and initializing string variables, reading strings from terminal, writing strings to screen, Arithmetic operations on characters, String handling functions - strlen, strcmp, strcpy, strstr and strcat; Character handling functions - toascii, toupper, tolower, isalpha, isnumeric etc.	Book 1	Chapter 8
User-defined functions: Need for user-defined functions, Declaring, defining and calling C functions, return values and their types, Categories of functions: With/without arguments, with/without return values. Nesting of functions. **Recursion:** Definition, example programs. **Storage classes:** Automatic, Global, Static, Register.	Book 1	Chapter 9
UNIT IV		
Pointers: Understanding pointers, accessing the address of a variable, declaring and initializing pointers, accessing a variable through its pointer, pointer expression, pointer increments and scale factor, pointers and arrays, pointer and strings.	Book 1	Chapter 11
Structures and unions: Structure definition, giving values to members, structure initialization, comparison of structure variables, arrays of structures, arrays within structures, Structure and functions, structures within structures. Unions.	Book 1	Chapter 10
File Handling in C: Create in Read/Write and Append mode, copying file.	Book 1	Chapter 12
The Pre-processor: Macro substitution, file inclusion.	Book 1	Chapter 14

Text Book:

1. E. Balagurusamy, Programming in ANSI C, 7th Edition, Tata McGraw Hill

Reference Books:

1. Herbert Schildt, C: The Complete Reference, 4th Edition.
2. Brain W. Kernighan, C Programming Language, 2nd Edition, Prentice Hall Software.
3. Kernighan & Ritchie: The C Programming Language, 2nd Edition, PHI.
4. Kamthane, Programming with ANSI and TURBO C, Pearson Education.
5. V. Rajaraman, Computer Programming in C, 2nd Edition, PHI.
6. S. Byron Gottfried, Programming with C, 2nd Edition, TMH.
7. Yashwant Kanitkar, Let us C, 15th Edition, BPB P.B. Kottur, Computer Concepts and Programming in C, 23rd Edition, Sapna Book House

Program Name	**BCA**	Semester	**I**
Course Title	**Discrete Mathematics for Computer Applications (Theory)**		
Course Code:	**BCCACS103**	No. of Credits	**05**
Contact hours	**5 Hours per week**	Duration of SEA/Exam	**3 Hours**
Formative Assessment Marks	**20**	Summative Assessment Marks	**80**

Course Outcomes (COs):

After the successful completion of the course, the student will be able to:

- Study and solve problems related to connectives, predicates and quantifiers under different situations.
- Understand the basic concepts of Discrete Probability.
- To develop the knowledge about derivatives and know various applications of differentiation.
- Understand the Applications of Discrete Mathematics in Modelling Computation. Understand the basic concepts of Mathematical reasoning, set and functions

Topics	Book	Sections / Sub sections
UNIT I		
Mathematical logic: Introduction, statements, Connectives, negation, conjunction, disjunction, statement formulas and truth tables, Examples 1,2,3, Exercises 1-2.4 (1,2,3,4), Conditional and Biconditional statements (Exclude program pg no.19), Examples (1,2,3,4,5), Exercises 1-2.6 (2,4), Tautology and contradiction, Exercise 1-2.8-1, equivalence of formulas, Example 1, duality law, Example 1 [No theorem], Tautological Implications [No theorem], Exercise 1-2.11 (1,2,5) [Exclude all theorems with proofs and algorithms in each Subsection]	Book - 1	1-1, 1-2: 1.2.1 to 1.2.11 (Exclude 1-2.5, 1-2.7)
Predicates and Quantifiers (Page No. 80-85), arguments, joint Daniel.	Book - 1	1-5.1, 1-5.2
Sets: Definition, Basic concepts, notation, inclusion and equality of sets, the power set, Family of sets. Exercise 2-1.3 (1, 2, 4), Page No 104-111, (exclude definition 2-1.7) Operations on sets (All definitions with no proof), Example 1, 3, 5, Exercise 2-1.4 (2, 7), (Page No 111-115), Venn diagrams, Exercise 2-1.5- 2, Ordered pairs, and n-tuples, Cartesian product, example 1, 2, Exercise 2-1-3,4 (Page No. 122-126)	Book - 1	2-1 2-1.1 to 2-1.9 (Exclude 2-1.6,2-1.7)

Relations: Introduction, Example 1, Exercise 2-3.1-1 (Page No.148-151,153), Properties of a binary relation in a set, Exercise 2-3.2-5, Example 1, 2, 3, Relation matrix and graph of a relation, (Page No 154-159), equivalence relations, Example 1,2 (Page no. 164-165), compatibility relations, composition of Binary relation Example 1,2,3,4, (Page No 176-180)	Book - 1	2.3, 2-3.1, 2-3.2, 2-3.3, 2-3.5, 2-3.6, 2-3.7 (Exclude definition 2-3.10, 2-3.12, 2-3.15, algorithm, and theorem 2-3.1, 2-3.2)
UNIT II		
Partial Ordering: Definitions, lexicographic ordering, partially ordered set, Hasse diagram, Example 1, 2(a, b, c), 3, well ordered set (definition 2-3.20), Exercise 2-3.9-1, (Page No. 183-189,191) **Function**s: Definition and introduction (except definition 2-4.2), Exercise 2-4.1-4,5, types of functions, composition of functions, Example 1, 2, Inverse functions Example 1, 2, Exercise 2-4.3-4 (Page No 192-205) (Only Theorem statements [no proofs]) **Counting**: Basics of counting, (Product rule, sum rule, the inclusion-exclusion principle), Example 1 to 5, 12, 13, 18, 19, Exercise- 1, 2, 3, Pigeon hole principle, (Only Theorem-1 statement), Example 1, 2, 3, Permutation and combination, Example 1, 2, 3, 4, 5, 10, 12, 13, Exercise-1,4, (only theorem and corollary statements- no proof), Generalized Permutations and Combinations, Example 1, 2, 3, 4, Theorem 1 and 2 (only statements) (only theorem and corollary statements- no proof), Generating permutation and combination, Example 1 to 5, inclusion and exclusion, Example 1, 2, 3	Book - 1 Book - 1 Book - 2	2-3.8, 2-3.9 2-4.1 to 2-4.3 6-1, 6-2, 6-3, 6.5. 6.6
UNIT III		
Discrete Probability: Introduction, finite probability, Example 1, 2, 4-6, (only theorem statements-no proof), probabilities of complements and unions of events (except probability reasoning), Example 8, 9, probability theory, Example 1, 2, conditional probability, Example 3, 4, independence, Example 5, 6, 7 (except pairwise and mutual independence), Bernoulli Trials and the Binomial Distribution), Random variables, Example 10, 11, (Except Monte Carlo Algorithm and probabilistic method), Baye's theorem(no proof) , Example 2, 3, 4, Expected value and variance, Example 1, 2, 3, (Except Linearity of Expectations, Average-Case Computational Complexity, The Geometric Distribution), Independent random variables, Example 11, 13,15 ,16 (except Chebyshevs inequality)	Book - 2	7.1, 7.2, 7.3, 7.4

Mathematical Induction: Induction, principle of mathematical induction, Example 1, proving inequalities, Example 5, 6, strong induction and well ordering (only statements).	Book - 2	5.1, 5.2
Number Theory: Division algorithm, Example 1, 3, 4, 5 ,7 Theorem 2, 3, Modular arithmetic, Primes and greatest common divisors, Definitions 1, 2, 3, 5, Least common multiples, Example 1, 4, 10, 11, 12, 15, The Euclidean algorithm, Example 16, Exercise 24, 32 (a, b, c), (No theorems, lemma, corollary and proofs)	Book - 2	4.1, 4.3
UNIT IV		
Graphs: Graphs and Graph models (Only definitions with example figures, Graph Terminology and Special Types of Graphs, Examples 1, 3, 4, 5, 6, 7, (only theorem statements, no proof) (Except Bipartite Graphs, Some Applications of Special Types of Graphs), Example 18, 19, Representing Graphs and Graph Isomorphism, Example 1, 2, 3, 4, 5, 6, 7, 8, Exercise 1, 3, 5, 7, 10, 13, Connectivity, Definition 1, 2, 3, 4, 5, Example 1, 4, 10 (Except all theorems and proofs of this section), Euler and Hamilton Paths, Definition 1, 2, Example 1,2,5, (except necessary and sufficient conditions for Euler circuits and paths, exclude all theorems and proofs of this section), Shortest-Path Problems, Planar Graphs-Introduction , all examples and Applications, Graph Coloring, Definition 1, 2, Theorem 1(no proof), Example 1, Exercise 1,3 5,7.	Book - 2	10.1 to 10.8
Trees: Directed tree, leaf node, branch node, ordered tree, degree of a node, forest, descendent, m-ary tree, conversion of directed tree into a binary tree. (Page No 494-500)	Book - 2	5-1.4
Applications of Discrete Mathematics in Modelling Computation: Language and Grammars – Introduction, Phrase-Structured, Types, Derivation Trees; Finite State Machines with Output – Introduction, Finite State Machines, Types; Finite State Machines without Output - Introduction, Set of Strings, Finite State Automata, Language Recognition by FSM; Language Recognition – Introduction; Turing Machine – Introduction, Definition.	Book - 2	Chapter 13 (complete)

Text Books:

1. J.P. Trembley and R. Manobar, Discrete Mathematical Structures, McGraw Hill Education Private Limited, New Delhi,
2. Kenneth H. Rosen, Discrete Mathematics and Its Applications, Eighth Edition, 2012.

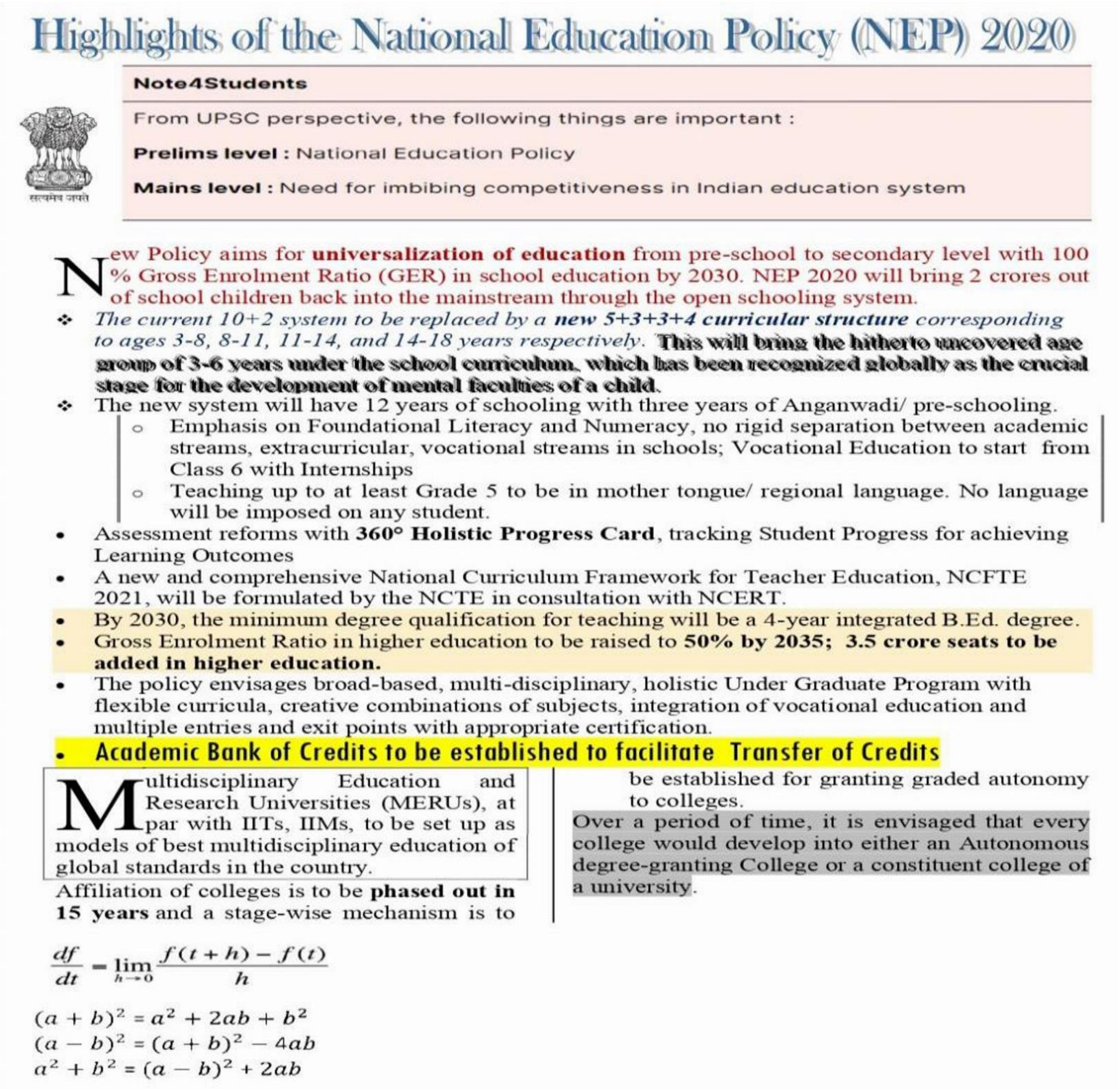
Reference Books:

1. C L Liu and D P Mohapatra, Elements of Discrete Mathematics- A Computer Oriented Approach, Fourth Edition, McGraw Hill Education Private Limited, New Delhi,
2. D C Sanchethi and V K Kapoor, Business Mathematics, Eleventh Revised Edition, Sulthan Chand & Sons Educational publishers, New Delhi,
3. Narsingh Deo, Graph Theory with Applications to Engg and Comp. Sci, PHI, 1986.
4. Ralph P. Grimaldi, B. V. Ramatta, Discrete and Combinatorial Mathematics, 5th Edition, Pearson, Education
5. K Chandrashekhara Rao, Discrete Mathematics, Narosa Publishing House, New Delhi

Program Name	**BCA**	Semester	**I**
Course Title	**Information Technology Lab**		
Course Code:	**BCCAPS101**	No. of Credits	**02**
Contact hours	**4 Hours per week**	Duration of SEA/Exam	**3 Hours**
Formative Assessment Marks	**10**	Summative Assessment Marks	**40**

PART -A: MS WORD

1. Prepare a document using different formatting tools.



Highlights of the National Education Policy (NEP) 2020

Note4Students

From UPSC perspective, the following things are important :

Prelims level : National Education Policy

Mains level : Need for imbibing competitiveness in Indian education system

New Policy aims for **universalization of education** from pre-school to secondary level with 100 % Gross Enrolment Ratio (GER) in school education by 2030. NEP 2020 will bring 2 crores out of school children back into the mainstream through the open schooling system.

- *The current 10+2 system to be replaced by a **new 5+3+3+4 curricular structure** corresponding to ages 3-8, 8-11, 11-14, and 14-18 years respectively.* **This will bring the hitherto uncovered age group of 3-6 years under the school curriculum, which has been recognized globally as the crucial stage for the development of mental faculties of a child.**
- The new system will have 12 years of schooling with three years of Anganwadi/ pre-schooling.
  - Emphasis on Foundational Literacy and Numeracy, no rigid separation between academic streams, extracurricular, vocational streams in schools; Vocational Education to start from Class 6 with Internships
  - Teaching up to at least Grade 5 to be in mother tongue/ regional language. No language will be imposed on any student.
- Assessment reforms with **360° Holistic Progress Card**, tracking Student Progress for achieving Learning Outcomes
- A new and comprehensive National Curriculum Framework for Teacher Education, NCFTE 2021, will be formulated by the NCTE in consultation with NCERT.
- By 2030, the minimum degree qualification for teaching will be a 4-year integrated B.Ed. degree.
- Gross Enrolment Ratio in higher education to be raised to **50% by 2035; 3.5 crore seats to be added in higher education.**
- The policy envisages broad-based, multi-disciplinary, holistic Under Graduate Program with flexible curricula, creative combinations of subjects, integration of vocational education and multiple entries and exit points with appropriate certification.
- Academic Bank of Credits to be established to facilitate Transfer of Credits

Multidisciplinary Education and Research Universities (MERUs), at par with IITs, IIMs, to be set up as models of best multidisciplinary education of global standards in the country.

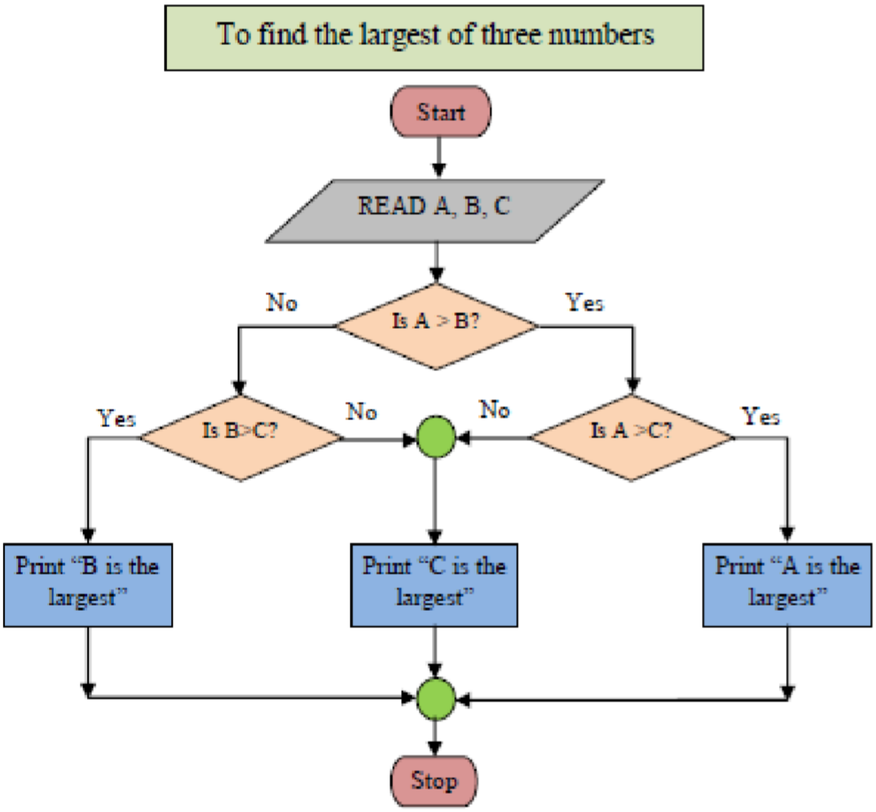
Affiliation of colleges is to be **phased out in 15 years** and a stage-wise mechanism is to be established for granting graded autonomy to colleges.

Over a period of time, it is envisaged that every college would develop into either an Autonomous degree-granting College or a constituent college of a university.

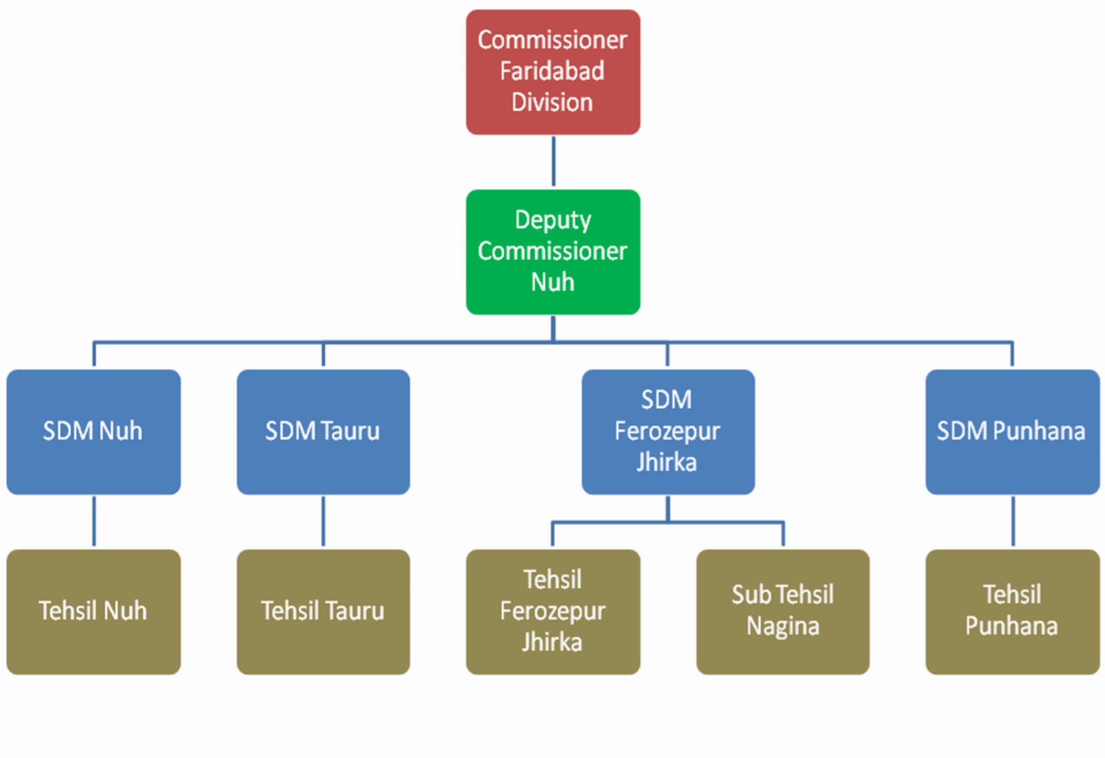
$$\frac{df}{dt} = \lim_{h \to 0} \frac{f(t+h) - f(t)}{h}$$

$$(a+b)^2 = a^2 + 2ab + b^2$$
$$(a-b)^2 = (a+b)^2 - 4ab$$
$$a^2 + b^2 = (a-b)^2 + 2ab$$

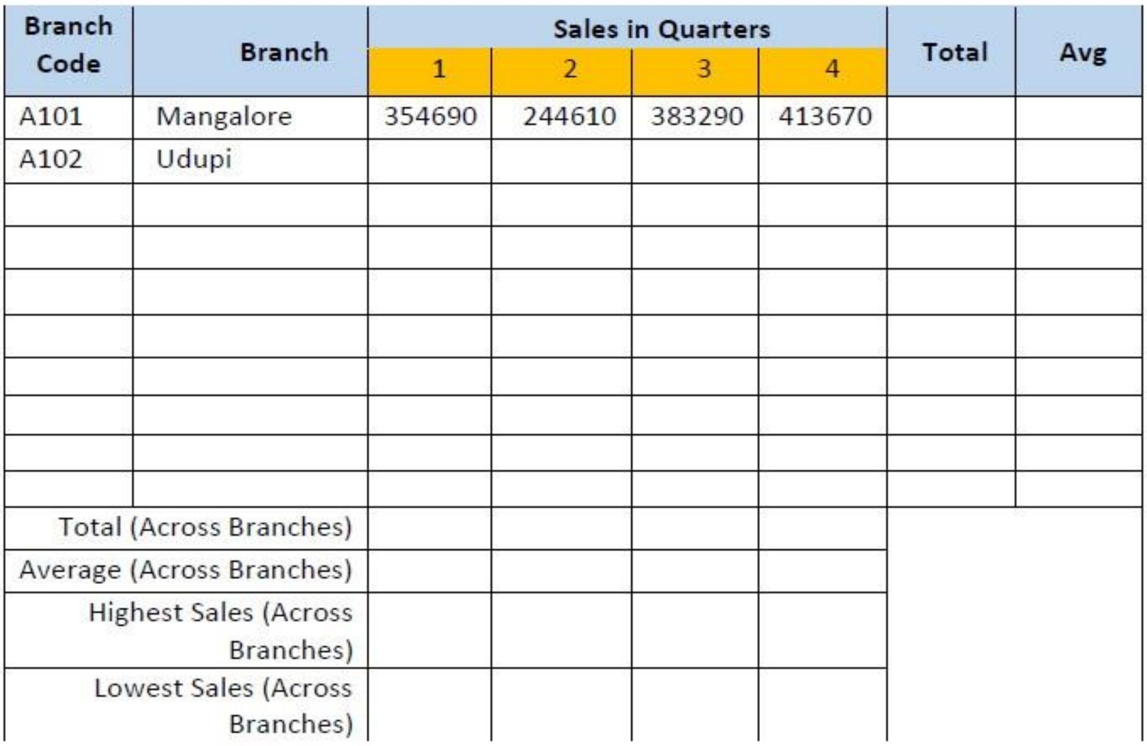
2. Prepare a document using SmartArt and Shapes tools.



Organization Chart – Administration Faridabad Division.

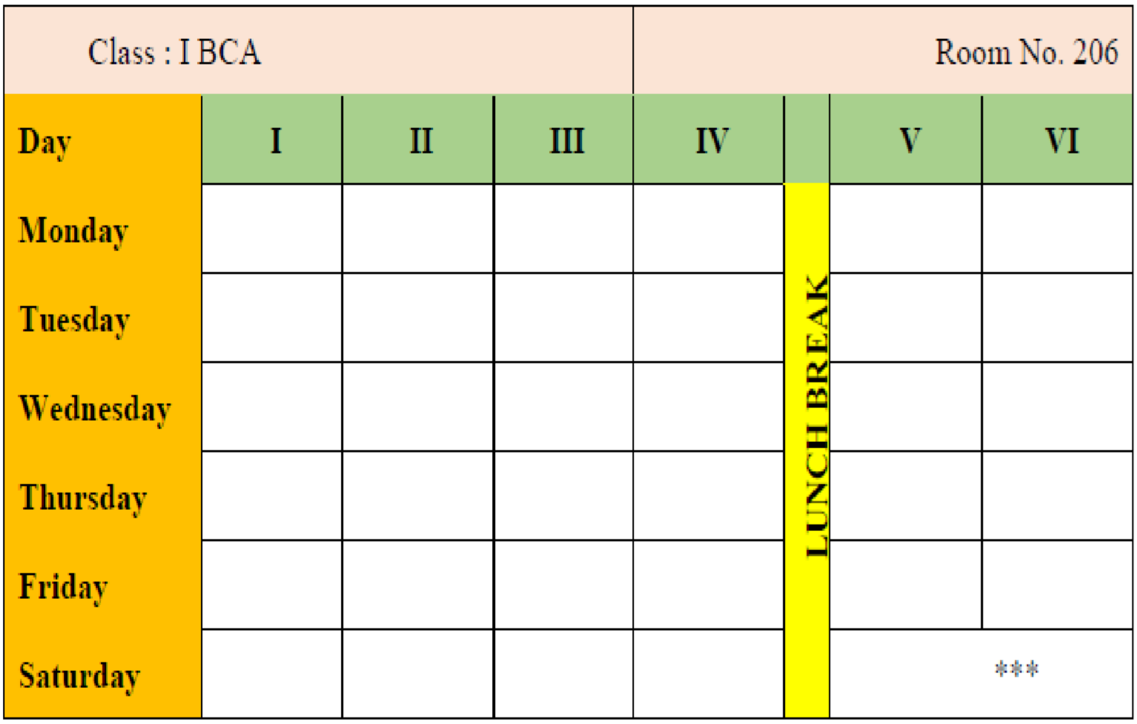


3. Prepare a document with table to store sales details of a company for different quarters and calculate total, average and find maximum, minimum sales value.



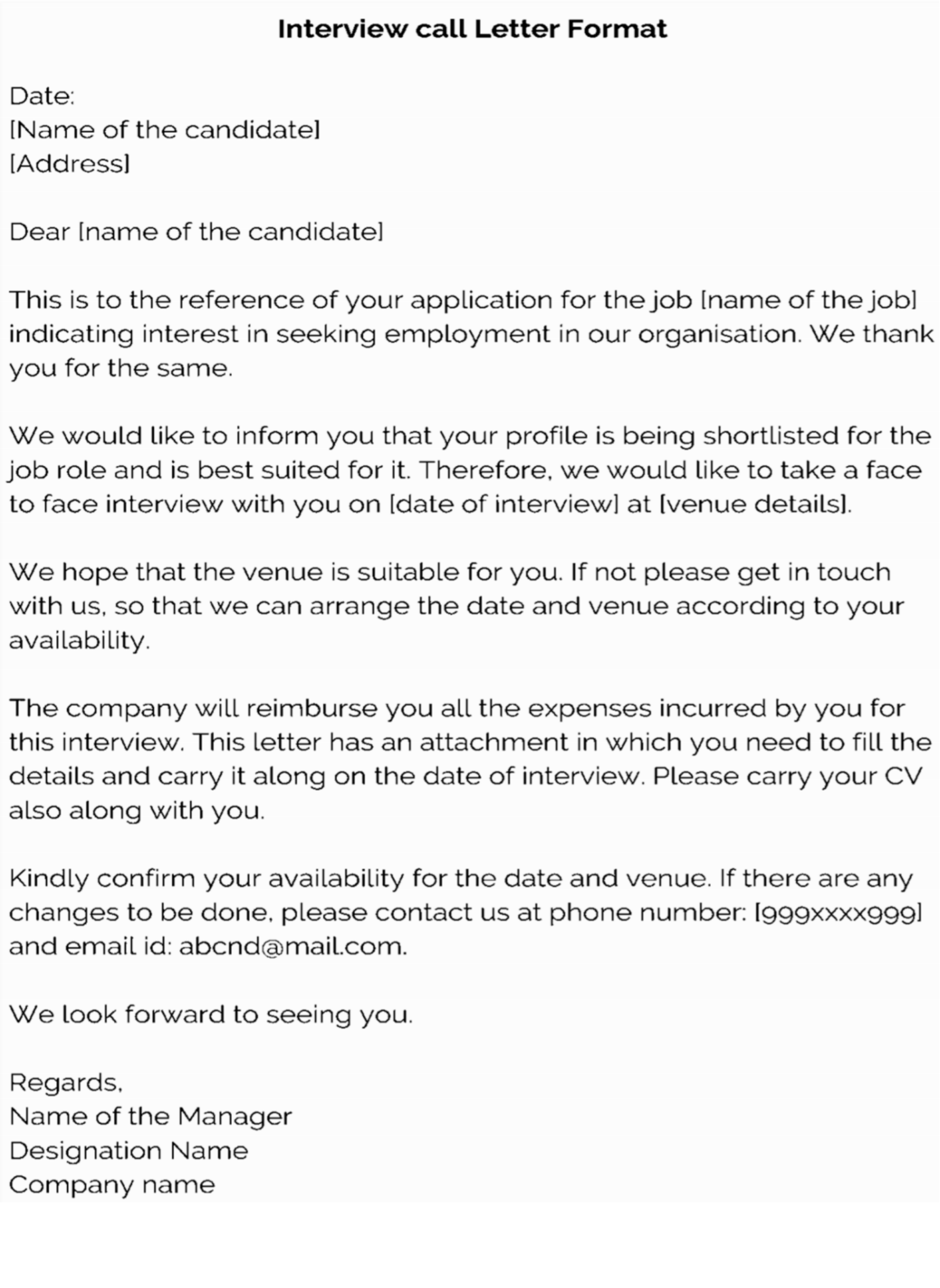
Branch Code	Branch	Sales in Quarters				Total	Avg
		1	2	3	4		
A101	Mangalore	354690	244610	383290	413670		
A102	Udupi						
Total (Across Branches)							
Average (Across Branches)							
Highest Sales (Across Branches)							
Lowest Sales (Across Branches)							

TIME TABLE



Class : I BCA				Room No. 206			
Day	I	II	III	IV		V	VI
Monday					LUNCH BREAK		
Tuesday							
Wednesday							
Thursday							
Friday							
Saturday						***	

4. Prepare interview call letters for five candidates describing about the company and instructions about the interview. Use Mail merge feature.



Interview call Letter Format

Date:
[Name of the candidate]
[Address]

Dear [name of the candidate]

This is to the reference of your application for the job [name of the job] indicating interest in seeking employment in our organisation. We thank you for the same.

We would like to inform you that your profile is being shortlisted for the job role and is best suited for it. Therefore, we would like to take a face to face interview with you on [date of interview] at [venue details].

We hope that the venue is suitable for you. If not please get in touch with us, so that we can arrange the date and venue according to your availability.

The company will reimburse you all the expenses incurred by you for this interview. This letter has an attachment in which you need to fill the details and carry it along on the date of interview. Please carry your CV also along with you.

Kindly confirm your availability for the date and venue. If there are any changes to be done, please contact us at phone number: [999xxxx999] and email id: abcnd@mail.com.

We look forward to seeing you.

Regards,
Name of the Manager
Designation Name
Company name

PART-B: MS POWERPOINT

1. Create a presentation (minimum 5 slides) about your college. It should contain images, chart, Bulleted text… The slides should be displayed automatically in a loop.

2. A simple quiz program. Use hyperlinks to move to another slide in the presentation to display the result and correct answer/wrong answer status. Use at least four questions. [**Navigation must be done by hyperlink**]

3. Create a presentation for a business proposal (minimum 5 slides).
   - Slides must include company logo in header
   - A title slide with table of contents
   - Financial data of the company in the table
   - Company sales and profit in charts
   - Make use of animation and transition

4. Create a presentation for a college project (minimum 5 slides).
   - Master slide
   - Add comments for each slide
   - Add Audio and video to the slide
   - Add header and footer
   - Add source citation
   - Make use of animation and transition

   [Presentation must include title slide, Module Design, Chart, references]

PART-C: MS EXCEL

(Note: Give proper titles, column headings for the worksheet. Insert 10 records for each exercise in such a way to get the result for all the conditions. Format the numbers appropriately wherever needed).

1. Create a worksheet to maintain student information such *as RollNo, Name, Class, Marks in three subjects* of 10 students. Calculate total marks, average and grade. Find grade for Distinction, First class, Second class, Pass and Fail using normally used conditions.
   - Using custom sort, sort the data according to class: - Distinction first, First Class next, and so on. Within each class, average marks should be in descending order.
   - Also draw the Column Chart showing the RollNo versus Average scored.

   (Note: Worksheet creation and formatting 3 marks, calculations: 3 marks, sorting: 2 marks, chart: 2 marks)

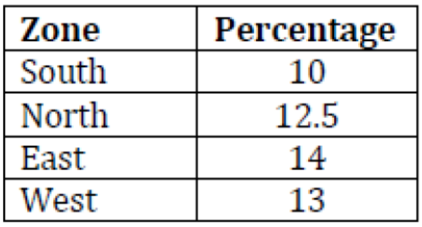
2. Prepare a worksheet to store details of electricity consumed by customers. Details are Customer No, Customer Name, Meter No, Previous meter reading, Current meter reading of 10 customers. Calculate total number of units consumed and total amount to be paid by each consumer using following conditions:
   - If unit consumed is up to 30, charge is 100.
   - 31 to 100 units, 4.70 per unit
   - 101 to 200 units, 6.25 per unit
   - Above 200 units, 7.30 per unit.
   - Use Data validation to see that current reading is more than previous reading.
   - Arrange the records in the alphabetic order of names.
   - Filter the records whose bill amount is more than Rs.1500.

   (Note: Worksheet creation and formatting 2 marks, Data validation: 2 marks, calculations: 2 marks, sorting: 2 marks, filtering: 2 marks)

3. Create Employee worksheet having EmpNo, EmpName, DOJ, Department, Designation and Basic Pay of 8 employees. Calculate DA, HRA, Gross Pay, Profession Tax, Net Pay, Provident Fund as per the rule:
   - DA = 30% of basic pay
   - HRA = 10% of basic pay if basic pay is less than 25000, 15% of basic pay otherwise.
   - Gross =DA +HRA+ Basic pay
   - Provident fund =12% of Basic pay or Rs.2000, whichever is less.
   - Profession Tax= Rs.100 if Gross pay is less than 10000, Rs.200 otherwise.
   - NetPay = Gross - (Professional tax + Provident Fund)
   - Using Pivot table, display the number of employees in each department and represent it using Pie chart.

   (Note: Worksheet creation and formatting 2 marks, calculations: 3 marks, Pivot table: 3 marks, Chart: 2 marks)

4. Create a table COMMISSION containing the percentage of commission to be given to salesmen in different zones as follows:



Zone	Percentage
South	10
North	12.5
East	14
West	13

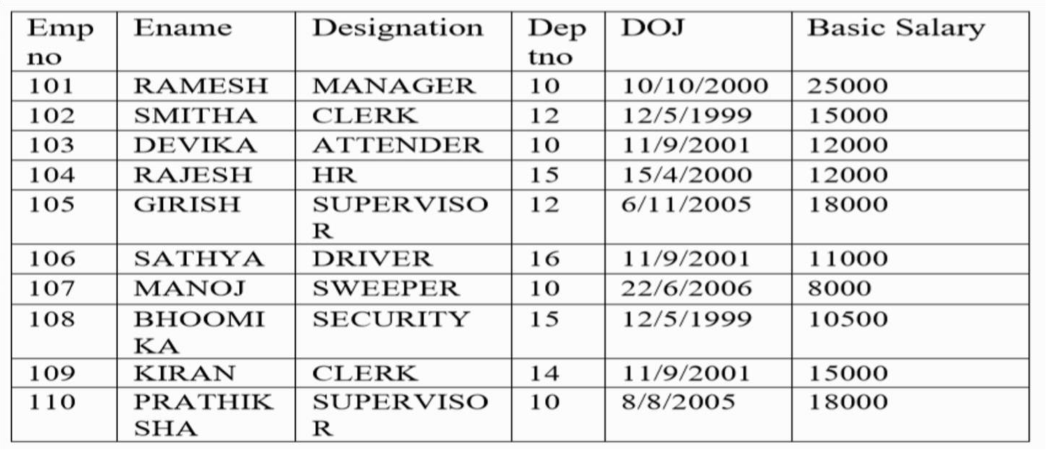
   Create another table SALES in the same worksheet to store salesman name, zone name, place, name of the item sold, rate per unit, quantity sold. Calculate total sales amount of each salesman. Referring the COMMISSION table, write the formula to compute the commission to be given. (Hint: Use if function and absolute cell addresses)

   Using advanced filtering show the result in other parts of the worksheet.
   - Show the records of various zones separately.
   - Show the records of only East and West zones.
   - Display the details of the items sold more than 50, in South or North zones.

   (Note: Worksheet creation and formatting: 2 marks, calculations: 2 marks, filtering: 6 marks)

PART-D: MS ACCESS

1. Create Employee database and table Emp using MS ACCESS with following Structure.

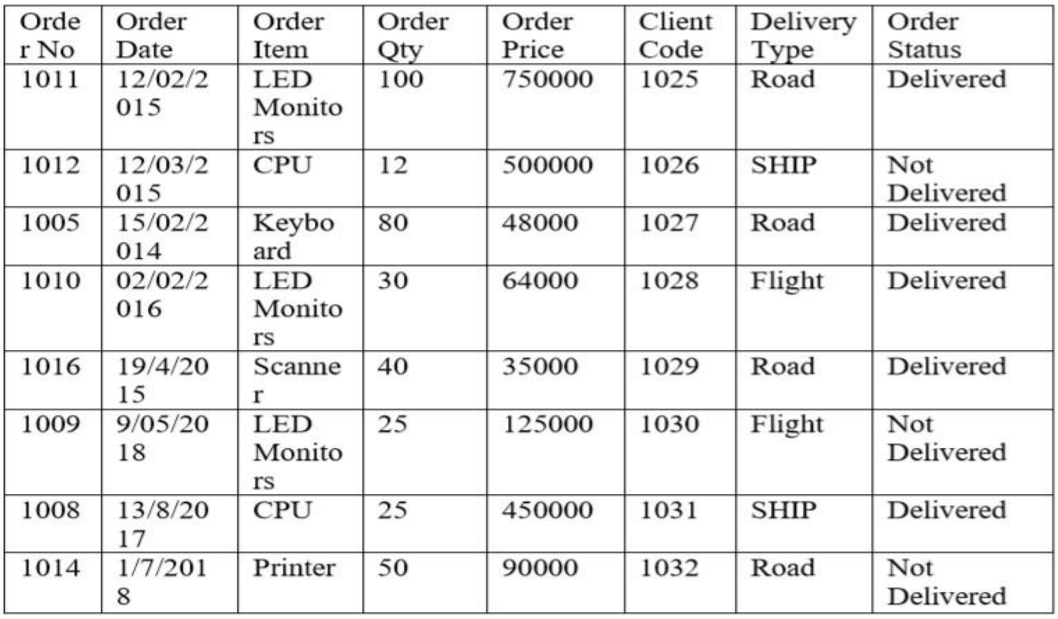


Emp no	Ename	Designation	Deptno	DOJ	Basic Salary
101	RAMESH	MANAGER	10	10/10/2000	25000
102	SMITHA	CLERK	12	12/5/1999	15000
103	DEVIKA	ATTENDER	10	11/9/2001	12000
104	RAJESH	HR	15	15/4/2000	12000
105	GIRISH	SUPERVISOR	12	6/11/2005	18000
106	SATHYA	DRIVER	16	11/9/2001	11000
107	MANOJ	SWEEPER	10	22/6/2006	8000
108	BHOOMIKA	SECURITY	15	12/5/1999	10500
109	KIRAN	CLERK	14	11/9/2001	15000
110	PRATHIKSHA	SUPERVISOR	10	8/8/2005	18000

Perform following operation:

a) List all the Employees who are working in Dept no.10
b) List all the Employees who get less than 20000 Salary
c) Update Salary by adding the increments as per the following:-
   i. 10% Increment in Basic Salary who get < 20000
   ii. 5% Increment in Basic Salary who get >=20000.

2. Create the “Order” database and a table “Orderdtl” having following records:

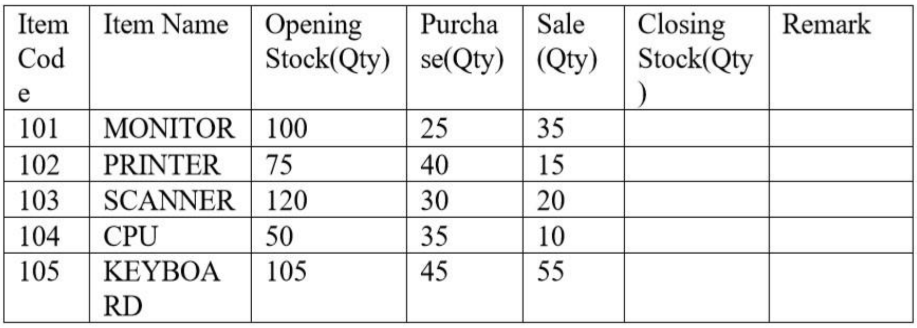


Order No	Order Date	Order Item	Order Qty	Order Price	Client Code	Delivery Type	Order Status
1011	12/02/2015	LED Monitors	100	750000	1025	Road	Delivered
1012	12/03/2015	CPU	12	500000	1026	SHIP	Not Delivered
1005	15/02/2014	Keyboard	80	48000	1027	Road	Delivered
1010	02/02/2016	LED Monitors	30	64000	1028	Flight	Delivered
1016	19/4/2015	Scanner	40	35000	1029	Road	Delivered
1009	9/05/2018	LED Monitors	25	125000	1030	Flight	Not Delivered
1008	13/8/2017	CPU	25	450000	1031	SHIP	Delivered
1014	1/7/2018	Printer	50	90000	1032	Road	Not Delivered

Execute following Query

a) Display all the Order No. which have not been yet Delivered.
b) Display all the Orders of LED Monitor and CPU.
c) Display all the Orders of LED Monitor and CPU which are not have been delivered yet.

3. Create a “Stock” database having “Inventory” table:



Item Code	Item Name	Opening Stock(Qty)	Purchase(Qty)	Sale (Qty)	Closing Stock(Qty)	Remark
101	MONITOR	100	25	35		
102	PRINTER	75	40	15		
103	SCANNER	120	30	20		
104	CPU	50	35	10		
105	KEYBOARD	105	45	55		

Execute following Query

a) Calculate the closing stock of each item (Closing Stock = Opening Stock + Purchase – Sales)
b) Display all the Items which has closing stock < 100
c) If closing stock is less than 100 then set the remark as “Re-Order Level” otherwise “Enough Stock”.

4. Create a “Company” database having “Sales” table with fields saleid, quarter, product, no_of_sales.

Perform the followings:

a) Design a form to insert records to Sales table
b) Generate a report to display Sales details of product based on quarters.

Evaluation Scheme for Lab Examination:

Assessment Criteria		
Program-1	**MS WORD**	**8 Marks**
Program-2	**MS POWERPOINT**	**7 Marks**
Program-3	**MS EXCEL**	**10 Marks**
Program-4	**MS ACCESS**	**10 Marks**
Practical Record		**05 Marks**
Total		**40 Marks**

Program Name	**BCA**	Semester	**I**
Course Title	**C Programming Lab**		
Course Code:	**BCCAPS102**	No. of Credits	**02**
Contact hours	**4 Hours per week**	Duration of SEA/Exam	**3 Hours**
Formative Assessment Marks	**10**	Summative Assessment Marks	**40**

PART – A

1. Program to find the roots of quadratic equation using else if ladder.
2. Program to read two integer values & an operator as character and perform basic arithmetic operations on them using switch case (+, -, *, / operations).
3. Program to reverse a number and find the sum of individual digits. Also check for palindrome.
4. Program to calculate and display the first ‘n’ Fibonacci numbers
5. Program to find given number is a prime or not.
6. Program to count occurrences of each character in a given string.
7. Program to read string with alphabets, digits and special characters and convert upper case letters to lower case and vice a versa and retain the digits and special characters as it is.
8. Program to search for number of occurrences of number in a list of numbers using one-dimensional array also display its positions.

PART-B

1. Program to find the largest and smallest elements with their position in a one-dimensional array.
2. Program to read ‘n’ integer values into a single dimension array and arrange them in ascending order using bubble sort method.
3. Menu driven Program to perform addition and multiplication of two Matrices.
4. Program to find nCr and nPr using recursive function to calculate factorial.
5. Program to read a string and count number of letters, digits, vowels, consonants, spaces and special characters present in it using user defined function.
6. Program sort a list of strings in ascending order using Pointers.
7. Program to enter the information of a student like name, register number, marks in three subjects into a structure and display total, average and grade Display details in a neat form.

8. Write a menu driven program to
   a. Create a text file
   b. Append the contents of a text file to another existing file by accepting filenames
   c. Display the content of entered filename
   d. Exit

   Create two text files during the execution of the program. Display their contents. Perform Appending. Display the contents again. Always check for the existence of the inputted file names.

Evaluation Scheme for Lab Examination:

Assessment Criteria		
Program-1	**PART-A** **Writing:7 Marks Execution:8Marks**	**15 Marks**
Program-2	**PART-B** **Writing:10 Marks Execution:10 Marks**	**20 Marks**
Practical Record		**05 Marks**
Total		**40 Marks**

CURRICULUM STRUCTURE FOR II SEMESTER BCA GENERAL

Semester II								
Sl. No.	**Course Code**	**Title of the Cour*se***	**Category of Courses**	**Teaching Hours per Week**	**SE E**	**IA**	**Total Marks**	**Credits**
1	**BCCACS201**	**Data Structures**	**Core**	**4**	**80**	**20**	**100**	**3**
2	**BCCACS202**	**Object Oriented Programming using Java**	**Core**	**4**	**80**	**20**	**100**	**3**
3	**BCCACS203**	**Computational Mathematics**	**Core**	**5**	**80**	**20**	**100**	**5**
4	**BCCAPS201**	**Data Structures Lab**	**Practical**	**4**	**40**	**10**	**50**	**2**
5	**BCCAPS202**	**Object Oriented Programming Lab**	**Practical**	**4**	**40**	**10**	**50**	**2**

Program Name	**BCA**	Semester	**II**
Course Title	**Data Structures (Theory)**		
Course Code	**BCCACS201**	No. of Credits	**03**
Contact hours	**4 Hours per week**	Duration of SEA/Exam	**3 Hours**
Formative Assessment Marks	**20**	Summative Assessment Marks	**80**

Course Outcomes (COs):

After the successful completion of the course, the student will be able to:

- Describe how arrays, records, linked structures, stacks, queues, trees, and graphs are represented in memory and used by algorithms
- Describe common applications for arrays, records, linked structures, stacks, queues, trees, and graphs
- Write programs that use arrays, linked structures, stacks, queues, trees, and graphs
- Demonstrate different methods for traversing trees
- Compare alternative implementations of data structures with respect to performance
- Describe the concept of recursion, give examples of its use
- Discuss the computational efficiency of the principal algorithms for sorting, searching, and hashing

Topics	Chapter	Page No / Section
UNIT 1 [13 HOURS]		
Introduction to data structures: Introduction, Basic terminology; Elementary Data Organization, Data Structures, Data Structure Operations	Chapter 1	1.1 – 1.4
Introduction to Algorithms, Preliminaries: Introduction, Algorithmic notations, Control structures.	Chapter 2	2.1, 2.3, 2.4
Recursion: Definition; Recursion Technique Examples – Factorial, Fibonacci sequence, Towers of Hanoi.	Chapter 6	6.8, 6.9 (complexity excluded)
Arrays: Basic Concepts – Definition, Declaration, Initialization, Operations on arrays, Types of arrays, Representation of Linear Arrays in memory, Traversing linear arrays, Inserting and deleting elements, Multidimensional arrays – Two Dimensional Arrays Representation of two – dimensional arrays, Sparse matrices.	Chapter 4	4.1, 4.2, 4.4, 4.5, 4.6, 4.10, 4.17

Sorting: Bubble sort Quick sort Insertion sort, Selection sort, Merge sort	Chapter 9 Chapter 4 Chapter 6 Chapter 9	9.1 4.7 6.7 9.3, 9.4, 9.6 (complexity excluded)
UNIT 2 [13 HOURS]		
Searching: Definition, Sequential Search, Binary search	Chapter 4	4.8, 4.9 (complexity excluded in both 4.8 & 4.9)
Dynamic memory management: Memory allocation and de-allocation functions - malloc, calloc, realloc and free. **Linked list:** Basic Concepts – Definition and Representation of linked list, Types of linked lists – Singly linked list, Doubly liked list, Header linked list, Circularly linked list, Representation of Linked list in Memory; Operations on Singly linked lists– Traversing, Searching, Insertion, Deletion, Memory allocation; Garbage collection	Chapter 4 Chapter 5	4.13 5.1 – 5.11
UNIT 3 [13 HOURS]		
Stacks: Basic Concepts –Definition and Representation of stacks, Array representation of stacks, Linked representation of stacks, Operations on stacks, Applications of stacks, Infix, postfix and prefix notations, Conversion from infix to postfix using stack, Evaluation of postfix expression using stack, Application of stack in function calls.	Chapter 6	6.1, 6.2, 6.3, 6.4, 6.5, 6.6, 6.7, 6.10
Queues: Basic Concepts – Definition and Representation of queues, Array representation of Queues, Linked representation of Queues, Types of queues - Simple queues, Circular queues, Double ended queues, Priority queues, Operations on queues	Chapter 6	6.11, 6.12, 6.13, 6.14, 6.15, 6.16 (6.79, 6.80, 6.87 only)
UNIT 4 [13 HOURS]		
Trees: Definition, Tree terminologies – node, root node, parent node, ancestors of a node, siblings, terminal & non-terminal nodes, degree of a node, level, edge, path, depth **Binary tree:** Type of binary trees - strict binary tree, complete binary tree, binary search tree; Array representation of binary tree, Traversal of binary tree-preorder, inorder and postorder traversal	Chapter 7	7.1, 7.2, 7.3, 7.4, 7.5, 7.8, 7.9
Graphs: Terminologies, Matrix representation of graphs; Traversal: Breadth First Search and Depth first search.	Chapter 8	8.1, 8.2, 8.3, 8.5, 8.7

Text Books:

1. Seymour Lipschutz, Data Structures with C, Schaum's Outlines Series, Tata McGraw Hill, 2011

Reference Books:

1. Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures, Computer Science Press, 1982.
2. Aaron M. Tenenbaum , Data structures using C, First Edition, Pearson Education
3. Kamathane, Introduction to Data structures, Pearson Education , 2004
4. Y. Kanitkar, Data Structures Using C, Third Edition, BPB
5. Padma Reddy: Data Structure Using C, Revised Edition 2003, Sai Ram Publications.
6. Sudipa Mukherjee, Data Structures using C – 1000 Problems and Solutions, McGraw Hill Education, 2007
7. R. Venkatesan and S. Lovelyn Rose, Data Structures, First Edition: 2015, Wiley India Pvt. Ltd. Publications

Program Name	**BCA**	Semester	**II**
Course Title	**Object Oriented Programming using Java (Theory)**		
Course Code:	**BCCACS202**	No. of Credits	**03**
Contact hours	**4 Hours per week**	Duration of SEA/Exam	**3 Hours**
Formative Assessment Marks	**20**	Summative Assessment Marks	**80**

Course Outcomes (COs):

After the successful completion of the course, the student will be able to:

- Understand the features of Java and the architecture of JVM
- Write, compile, and execute Java programs that may include basic data types and control flow constructs and how type casting is done
- Identify classes, objects, members of a class and relationships among them needed for a specific problem and demonstrate the concepts of polymorphism and inheritance
- The students will be able to demonstrate programs based on interfaces and threads compared to other Programming Language
- Write, compile, execute Java programs that include GUIs and event driven programming and also programs based on files

Topics	Book	Chapter / Page No / Section
UNIT 1 [13 HOURS]		
Fundamentals of Object Oriented Programming: Introduction, Object Oriented Paradigm, Basic Concepts of OOP, Benefits and Applications of OOP.	Book 1 Chapter 1	1.1 – 1.5
Introduction to Java: Java Features, Java Environment, Simple Java Program, Java Program Structure, Java Tokens, Java Statements, Java Virtual Machine.	Book 1 Chapter 2 Chapter 3	2.2, 2.9 3.2, 3.5, 3.6, 3.7, 3.9
Java Programming Basics: Constants, Variables, Data Types, Declaration of variables, Giving values to the variable, Scope of variables, Symbolic constants, Type casting.	Book 1 Chapter 4	4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9
Operators and Expressions: Arithmetic Operators, Relational Operators, Logical Operators, Assignment Operator, Increment and Decrement Operators, Conditional Operator, Special Operators, Mathematical functions.	Book 1 Chapter 5	5.1, 5.2. 5.3, 5.4, 5.5, 5.6, 5.7, 5.9, 5.15

Using I/O: Byte streams and character streams, predefined streams, reading console input, reading characters, strings, writing console output.	Book 2 Chapter13	Pg Nos. 285, 286. 288 – 292
Decision Making & Branching: Simple if statement, if..else statement, nesting of if..else statement, the else..if ladder, the Switch statement.	Book 1 Chapter 6	6.1 – 6.7
UNIT 2 [13 HOURS]		
Decision making & Looping -The while statement, the do statement, the for statement. Jumps in loops, Labelled loops.	Book 1 Chapter 7	7.1 – 7.6
Class & Objects - Class Fundamentals, Declaring Objects, Assigning Object Reference Variables, Introducing Methods, Constructors, The 'this' keyword,	Book 2 Chapter 6	Pg. Nos. 105 – 120
Overloading Methods, Using Objects as Parameters, Returning Objects, Recursion, Understanding 'static', Introducing 'final', Using Command-Line Arguments, Varargs : Variable-Length Arguments	Book 2 Chapter 7	Pg. Nos. 125 – 132, 134 – 136, 141 – 143, 150 – 152
Arrays and Strings: One dimensional array, Creating an array, Two dimensional arrays, Strings, Vectors, Wrapper classes.	Book 1 Chapter 9	9.1 – 9.7
UNIT 3 [13 HOURS]		
Inheritance - Inheritance Basics, Using 'super', Creating Multilevel hierarchy, Method Overriding, Using Abstract Classes, Using final with Inheritance.	Book 2 Chapter 8	Pg. Nos. 157 – 173, 177 – 180
Packages & Interfaces - Packages, Access protection in packages, Importing Packages, Interfaces.	Book 2 Chapter 9	Pg. Nos. 183 – 194
Exception Handling - Exception Handling Fundamentals –Exception Types, Uncaught Exceptions, Using try and catch, Multiple catch clauses, finally, Java's built-in Exceptions	Book 2 Chapter 10	Pg. Nos. 205 – 210, 216 – 218
Multithreaded Programming - Introduction, Creating threads, Extending the thread class, stopping & blocking thread, Life cycle of a thread, Using thread methods, Implementing the runnable interface.	Book 1 Chapter 12	12.1 – 12.6, 12.10
UNIT 4 [13 HOURS]		
Event and GUI programming: The Applet Class, Types of Applets, Applet Basics, Applet Architecture, An Applet	Book 2 Chapter 21	Pg. Nos. 617 = 625,

Skeleton, Simple Applet Display Methods, Requesting Repaint, The HTML APPLET tag.		629 – 630
Event Handling: The delegation event model, Event Classes ActionEvent, KeyEvent & MouseEvent Classes, Event Listener Interfaces –ActionListener, KeyListener & MouseListener interfaces. Using the Delegation Event Model.	Book 2 Chapter 22	Pg. Nos. 637 – 641, 645, 646, 650 – 658
Window Fundamentals: Working with Frame Windows, Creating a Frame Window in an Applet. Creating a Windowed Program, Displaying information within a window.	Book 2 Chapter 23	Pg. Nos. 666 – 676
Introducing swing – two key swing features, components and containers, the swing packages, a simple swing application, event handling.	Book 2 Chapter 29	Pg. Nos. 859, 860, 862 – 865, 868
Exploring Swing: JLabel, JTextField, JButton, Checkboxes, Radio buttons, JList, JComboBox.	Book 2 Chapter 30	Pg. Nos. 879 – 885, 887 – 891, 895 – 900

Text Books:

1. E Balagurusamy, Programming with Java – A Primer, Fourth Edition, Tata McGraw Hill Education Private Limited.
2. Herbert Schildt, Java: The Complete Reference, Seventh Edition, McGraw Hill Publication.

Reference Books:

1. Herbert Schildt, Java 2-The Complete Reference,Fifth Edition, McGraw Hill publication.
2. CayS. Horstmann, Core Java VolumeI–Fundamentals, Prentice Hall.
3. Somashekara, M.T., Guru, D.S., Manjunatha, K.S, Object Oriented Programming with Java, EEE Edition, PHI.

Program Name	**BCA**	Semester	**II**
Course Title	**Computational Mathematics (Theory)**		
Course Code:	**BCCACS203**	No. of Credits	**05**
Contact hours	**5 Hours per week**	Duration of SEA/Exam	**3 Hours**
Formative Assessment Marks	**20**	Summative Assessment Marks	**80**

Course Outcomes (COs):

After completing this course satisfactorily, a student will be able to:

- Obtain an intuitive and working understanding of numerical methods for the basic problems of numerical analysis.
- Gain experience in the implementation of numerical methods using a computer.
- Trace error in these methods and need to analyse and predict it.
- Provide knowledge of various significant and fundamental concepts to inculcate in the students an adequate understanding of the application of Statistical Methods.
- Demonstrate the concepts of numerical methods used for different applications

Topic	Chapter	Section
UNIT I		
Errors in Numerical Computation – Errors and their computation.	Chapter 1	Explanation - 1.4 (Two types of errors), 1.4.1 (Formula of Absolute, Relative and Percentage errors). Examples – 1.2, 1.3, 1.5
Solution of Algebraic and Transcendental equations – Introduction, The Bisection method, The Iterative method, The method of False position, Newton-Raphson method, Ramanujan's method.	Chapter 2	Explanation – 2.1, 2.2 (Examples – 2.1, 2.2) Explanation – 2.3 (Excluding theorem), (Examples – 2.5, 2.6) Explanation – 2.4, (Example – 2.8) Explanation 2.5 (till equation 2.22), (Examples – 2.9, 2.10). 2.6 (Explanation only)

Interpolation – Introduction,	Chapter 3	Definitions of Interpolation and Extrapolation – 3.1. Explanation
Finite differences,	Chapter 3	Explanation – 3.3
Forward differences,		Explanation – 3.3.1
Backward differences,		Explanation – 3.3.2
Central differences,		Explanation – 3.3.3
Newton’s formula for interpolation,		Derivation – 3.6 (Examples – 3.4, 3.5, 3.6, 3.7) (Exercise – 5, 7, 8).
Lagrange’s interpolation formula,		Explanation – 3.9.1 (Derivation, Examples – 3.15, 3.17) (Exercise 25, 26, 30)
Divided differences,		Explanation – 3.11 (Only equations 3.102, 3.104 and 3.105)
Newton’s general interpolation formula.		Derivation – 3.11.1 (Examples 3.23, 3.24)
UNIT II		
Least Squares – Introduction, Least squares curve fitting procedures, Fitting a straight line,	Chapter 4	Explanation – 4.1, 4.2, 4.2.1, 4.2.2 (Examples – 4.1, 4.2, 4.3, 4.4)
Numerical Differentiation and Integration – Numerical Differentiation,	Chapter 5	Explanation – 5.1, Derivation – 5.2 (Examples – 5.1, 5.2, 5.3) (Exercise – 3, 5)
Numerical Integration,		Explanation – 5.4
Trapezoidal Rule,		Explanation – 5.4.1 (Example – 5.7)
Simpson’s 1/3-Rule,		Explanation – 5.4.2 (Examples – 5.9, 5.11)
Simpson’s 3/8-Rule.		Explanation – 5.4.3 (Exercise – 10, 11, 12)
UNIT III		
Matrices and Linear System of equations –	Chapter 6	
Basic definitions,		Explanation – 6.2
Matrix operations,		Explanation – 6.2.1 (Example – 6.1)
Transpose of a Matrix,		Explanation – 6.2.2 (Examples – 6.2, 6.3, 6.4)

The Inverse of a Matrix,		Explanation – 6.2.3, (Example – 6.5)
Matrix norms	Chapter 6	Explanation – 6.2.6 (only Matrix norms), (Example – 6.9)
Solution of Linear System – Direct Methods		
Matrix Inversion Method		Derivation – 6.3.1 (Example – 6.10)
Gaussian Elimination Method		Derivation – 6.3.2 (Example – 6.11)
Gauss-Jordan Method		Derivation – 6.3.3 (Example – 6.12)
LU Decomposition		Derivation – 6.3.4 (Example – 6.13)
Solution of Linear System – Iterative Methods Gauss-Seidel Method, Jacobi's Method.		Explanation – 6.4 (Example – 6.19) (Exercise – 1, 2, 3, 5, 9, 10, 14(a))
UNIT IV		
Numerical Solution of Ordinary Differential Equations		
Solution by Taylor's Series	Chapter 7	Explanation – 7.2, (Example – 7.1)
Euler's Method		Explanation – 7.4, (Example – 7.4)
Modified Euler's Method		Explanation – 7.4.2, (Example – 7.6)
Runge-Kutta Method		Explanation – 7.5 (No derivation), (Example – 7.7)
Predictor-Corrector methods		Explanation – 7.6
Adams-Moulton Method		Derivation – 7.6.1, (Example – 7.10)
Milne's Method		Derivation – 7.6.2, (Example – 7.11)
Boundary Value Problems		Explanation – 7.10
Finite-Difference Method		Explanation – 7.10.1 (No Derivation), (Example – 7.13) (Exercise – 1, 6, 7, 10, 14, 20)

Text Book:

1. S.S. Sastry, Numerical Analysis, 3rd edition, PHI publication.

Reference Books:

1. M. K. Jain, S.R.K. Iyengar & R. K. Jain, Numerical methods for Scientific and Engineering computation, 5th edition, New Age International publishers.
2. V Rajaraman, Computer Oriented Numerical Methods, 3rd Edition, PHI, 2006

Program Name	**BCA**	Semester	**II**
Course Title	**Data Structures Lab**		
Course Code:	**BCCAPS201**	No. of Credits	**02**
Contact hours	**4 Hours per week**	Duration of SEA/Exam	**3 Hours**
Formative Assessment Marks	**10**	Summative Assessment Marks	**40**

PART – A

1. Program to sort the given list using selection sort technique.
2. Program to sort the given list using insertion sort technique.
3. Program to solve Tower of Hanoi using Recursion.
4. Program to sort the given list using merge sort technique.
5. Program to sort the given list using quick sort technique.
6. Program to search an element using recursive binary search technique.
7. Program to implement Stack operations using arrays.
8. Program to implement Queue operations using arrays.

PART – B

1. Program to implement circular queue using array.
2. Program to implement Stack operations using linked list.
3. Program to implement Queue operations using linked list.
4. Program to evaluate given postfix expression.
5. Program to convert the given infix expression to postfix expression.
6. Program to perform insert node at the end, delete a given node and display contents of single linked list.
7. Menu driven program for the following operations on Binary Search Tree (BST) of Integers
   a) Create a BST of N Integers
   b) Traverse the BST in Inorder, Preorder and Post Order
8. Program for the following operation on the graph (G) of cities
   a) Create a graph of N cities using Adjacency Matrix
   b) Print all the nodes reachable from a given starting node in a diagraph using BFS method

Evaluation Scheme for Lab Examination:

Assessment Criteria		
Program-1	PART-A Writing:7 Marks Execution:8 Marks	15 Marks
Program-2	PART-B Writing:10 Marks Execution:10 Marks	20 Marks
Practical Record		05 Marks
Total		40 Marks

Program Name	**BCA**	Semester	**II**
Course Title	**Object Oriented Programming Lab**		
Course Code:	**BCCAPS202**	No. of Credits	**02**
Contact hours	**4 Hours per week**	Duration of SEA/Exam	**3 Hours**
Formative Assessment Marks	**10**	Summative Assessment Marks	**40**

PART – A

1. Program to accept student name and marks in three subjects. Find the total marks, average and grade (depending on the average marks).
2. Program, which reads two numbers having same number of digits. The program outputs the sum of product of corresponding digits. (Hint Input 327 and 539 output 3x5+2x3+7x9=84)
3. Program to input Start and End limits and print all Fibonacci numbers between the ranges. (Use for loop)
4. Define a class named Pay with data members String name, double salary, double da, double hra, double pf, double grossSal, double netSal and methods: Pay(String n, double s) - Parameterized constructor to initialize the data members, void calculate() - to calculate the following salary components, and void display() - to display the employee name, salary and all salary components.

   Dearness Allowance = 15% of salary
   House Rent Allowance = 10% of salary
   Provident Fund = 12% of salary
   Gross Salary = Salary + Dearness Allowance + House Rent Allowance
   Net Salary = Gross Salary - Provident Fund

   Write a main method to create object of the class and call the methods to compute and display the salary details. [class basics]
5. Program to create a class DISTANCE with the data members feet and inches. Use a constructor to read the data and a member function Sum ( ) to add two distances by using objects as method arguments and show the result. (Input and output of inches should be less than 12.).
6. Program to create a class “Matrix” that would contain integer values having varied numbers of columns for each row. Print row-wise sum.
7. Program to extract portion of character string and print extracted string. Assume that ‘n’ characters extracted starting from mth character position.
8. Program to add, remove and display elements of a Vector.

PART – B

1. Create a class named 'Member' having data members: Name, Age, PhoneNumber, Place and Salary. It also has a method named 'printSalary' which prints the salary of the members. Two classes 'Employee' and 'Manager' inherit the 'Member' class. The 'Employee' and 'Manager' classes have data members 'specialization' and 'department' respectively. Now, assign name, age, phone number, address and salary to an employee and a manager by making an object of both of these classes and print the same. [inheritance]

2. Program to implement the following class hierarchy: Student: id, name
   StudentExam (derived from Student): Marks of 3subjects, total marks
   StudentResult (derived from StudentExam) : percentage, grade
   Define appropriate methods to accept and calculate grade based on existing criteria and display details of N students

3. Write a Program to calculate marks of a student using multiple inheritance implemented through interface. Class Student with data members rollNo, name, Stringcls and methods to set and put data.
   Create another class test extended by class Student with data members mark1, mark2, mark3 and methods to set and put data.
   Create interface sports with members sportsWt = 5 and putWt().

   Now let the class results extends class test and implements interface sports. Write a Java program to read required data and display details in a neat format.

4. Write a Program to create an abstract class named shape that contains two integers and an empty method named print Area().
   Provide three classes named Rectangle, Triangle and Ellipse such that each one of the classes extends the class shape. Each one of the class contains only the method print Area() that print the area of the given shape.[Abstract class].

5. Create a package to convert temperature in centigrade into Fahrenheit, and one more package to calculate the simple Interest. Implement both package in the Main () by accepting the required inputs for each application.

6. Write a Program that implements a multi-threaded program has three threads. First thread generates a random integer every second, and if the value is even, second thread computes the square of the number and prints. If the value is odd the third thread will print the value of cube of the number. [Multithreading]

7. Program that creates a user interface to perform basic integer operations.
   The user enters two numbers in the TextFields - Num1 and Num2. The result of operations must be displayed in the Result TextField when the "=" button is clicked. Appropriate Exception handling message to be displayed in the Result TextFieldwhen Num1 or Num2 is not an integer or Num2 is Zero when division operation is applied.

8. Using the swing components, design the frame for shopping a book that accepts book code, book name, and Price. Calculate the discount on code as follows.

Code	Discount rate
101	15%
102	20%
103	25%
Any other	5%

   Find the discount amount and Net bill amount. Display the bill.

Evaluation Scheme for Lab Examination:

Assessment Criteria		
Program-1	PART-A Writing:7 Marks Execution: 8Marks	15 Marks
Program-2	PART-B Writing:10 Marks Execution:10Marks	20 Marks
Practical Record		05 Marks
Total		40 Marks

Scheme of Assessment for Theory Examination

Question Pattern		Marks
Part - A		
1. Answer any ten sub - questions: (10 x 2 = 20)		20
Sub- question	Unit	
a, b, c	1	
d, e, f	2	
g, h, i	3	
j, k, l	4	
Part - B (Answer any ONE full question from each unit - 15 marks each)		
Unit - 1		15
2.		
3.		
Unit - 2		15
4.		
5.		
Unit - 3		15
6.		
7.		
Unit - 4		15
8.		
9.		
Total		80