ST. JOSEPH'S COLLEGE (AUTONOMOUS)

BENGALURU-27

DEPARTMENT OF COMPUTER SCIENCE

SYLLABUS FOR UNDERGRADUATE COURSE

Re-accredited with 'A++' GRADE and 3.79/4 CGPA by NAAC

Recognized by UGC as College of Excellence

To be implemented from 2018 -19 Batch

Course Structure and Syllabus for

Bachelor of Computer Applications (BCA)

(Six Semester Course)

Proposed for 2018 batch onwards

With

CHOICE BASED CREDIT SYSTEM (CBCS)

Department of Computer Science
Faculty of Science
St. Joseph's College (Autonomous)

DEPARTMENT OVERVIEW

VISION STATEMENT

The vision of the department is to make possible for our students to have a deep and analytical understanding of the field and to enable them to use their immense potential to enhance the quality of human life.

MISSION STATEMENT

The mission of the department is to offer a high-quality science education in the science of computing, as well as to prepare students for career opportunities in this area requiring a high level of technical knowledge and skill. The department will continue to develop and administer programs which serve the needs of industrial computer scientists, researchers, and computer application specialists for whom the discipline of computing sciences provides indispensable tools.

INTRODUCTION TO PROGRAM

Bachelor of Computer Applications is a three year course which consists of six semesters. It is an undergraduate programme. The latest technology is facilitated for the swift learning and ability to adapt to the challenging circumstances of the present era. The infrastructure provides a best ambience for the student to excel in the field of computers. The enhancement of the teaching-learning process happens to the maximum level in the order.

PROGRAM OBJECTIVE

Prepare men and women who confidently face the challenging world and work in the most advanced stages of computers. Facilitates overall understanding of the requirements from the subject point of view. Prepare the students to provide professional solutions to real time problems.

SEMESTER WISE COURSE STRUCTURE FOR FIRST YEAR BCA

FIRST SEMESTER

Theory							
Sl.No.	Code	Course Title	Hrs / Week	Credits	IA	SE	Total
1	-	English	4	4	30	70	100
2	-	Language	4	4	30	70	100
3	CA1118	Information Technology and Problem Solving	4	4	30	70	100
4	CA1218	Discrete mathematics	4	4	30	70	100
5	CA1318	Programming In C	4	4	30	70	100
6	CA1418	Digital Fundamentals and Logic Design	4	4	30	70	100
Practical							
1	CA1P1	C Programming Lab	2	1	70	30	100
2	CA1P2	Automation and digital logic lab	2	1	70	30	100

SECOND SEMESTER

Theory							
Sl.No.	Code	Course Title	Hrs / Week	Credits	IA	SE	Total
1	-	English	4	4	30	70	100
2	-	Language	4	4	30	70	100
3	CA2118	Data Structures Using C	4	4	30	70	100
4	CA2218	Microprocessors	4	4	30	70	100
5	Ca2318	Operating System	4	4	30	70	100
6	CA2418	Numerical methods and Statistics	4	4	30	70	100
Practical							
1	CA2P1	Data Structures Lab	2	1	70	30	100
2	CA2P2	Microprocessor Lab	2	1	70	30	100

THIRD SEMESTER

Theory							
Sl.No.	**Code**	**Course Title**	**Hrs / Week**	**Credits**	**IA**	**SE**	**Total**
1	-	English	4	4	30	70	100
2	-	Language	4	4	30	70	100
3	CA3118	Object Oriented Programming Using Python	4	4	30	70	100
4	CA3218	.NET Technologies	4	4	30	70	100
5	CA3318	Software Engineering	4	4	30	70	100
6	CA3418	Database Management Systems	4	4	30	70	100
Practical							
1	CA3P1	Python Programming Lab	2	1	70	30	100
2	CA3P2	.NET and Database Management Systems Lab	2	1	70	30	100

FOURTH SEMESTER

Theory							
Sl.No.	**Code**	**Course Title**	**Hrs / Week**	**Credits**	**IA**	**SE**	**Total**
1	-	English	4	4	30	70	100
2	-	Language	4	4	30	70	100
3	CA4118	JAVA Programming	4	4	30	70	100
4	CA4218	UNIX Programming	4	4	30	70	100
5	CA4318	Data Communication and Computer Networks	4	4	30	70	100
Open Electives (For students of other courses)							
6	CAOE4418	Web Development	2	2	15	35	50
7	CAOE4518	Problem Solving and Basic Programming Skills	2	2	15	35	50
Practical							
1	CA4P1	JAVA Programming Lab	2	1	70	30	100
2	CA4P2	UNIX Lab	2	1	70	30	100

FIFTH SEMESTER

Theory							
Sl.No.	**Code**	**Course Title**	**Hrs / Week**	**Credits**	**IA**	**SE**	**Total**
1	CA5118	Web Technologies	4	4	30	70	100
2	CA5218	System Software	4	4	30	70	100
3	CA5318	Computer Graphics and Multimedia	4	4	30	70	100
4	CA5415	Operations Research	4	4	30	70	100
Practical							
1	CA5P1	Web Technologies and Computer Graphics Lab	2	1	70	30	100
2	CA5P2	Mini Project Lab	2	1	70	30	100

SIXTH SEMESTER

Theory							
Sl.No.	**Code**	**Course Title**	**Hrs / Week**	**Credits**	**IA**	**SE**	**Total**
1	CA6118	Computer Organization and Architecture	4	4	30	70	100
2	CA6218	Object Oriented Analysis and Design using Unified Modeling Language	4	4	30	70	100
3	CA6318	Software Testing and Quality Assurance	4	4	30	70	100
4	CA6418	Mobile Applications	4	4	30	70	100
Practical							
1	CA6P1	Mobile Applications Lab	2	1	70	30	100
2	CA6P2	Major Project Lab	2	1	70	30	100

EXAMINATION AND ASSESMENTS

1. **IA Weightage** **30 %**
2. **End Semester Examination Weightage** **70%**

NOTE

The self study topics assigned in all the papers are within the syllabus and are included in the examination questions.

QUESTION PAPER PATTERN

The question papers of the theory examinations should follow the pattern specified below:

Section	Marks for each question	Number Of Questions		Total Marks
		Total	Should Answer	
A	2	10	10	20
B	6	7	5	30
C	10	3	2	20

Total Marks **70**

While selecting the questions importance should be given to all the units.

PRACTICAL QUESTION PAPER FORMAT

Scheme of valuation:

1.	Writing two programs one from each section	20 marks
2.	Execution of one program	20 marks
3.	Formatting the program and output	10 marks
3.	Record verification	10 marks
4.	Viva voce related to practical topics only	10 marks
	Total	70 marks

PROJECT EVALUATION FORMAT

Scheme of valuation:

1.	Demonstration and presentation	35 marks
2.	Documentation	35 marks
	Total	**70 marks**

INTERNAL ASSESSMENT FORMAT
THEORY

1.	IA test	15marks
2.	One Activity	10 marks
3.	Attendance	5 marks
	Total	**30 marks**

PRACTICALS

Every practical class the student should be assessed.

1.	Writing the observation book	3 marks
2.	Executing the programs	5 marks
3.	Record writing	2 marks
	Total	**10 marks**

PROJECT:

Internal marks for the final semester project work can be awarded by the guide by evaluating the performance of the student during the course of the project work.

30 marks

FIRST SEMESTER

Title: **INFORMATION TECHNOLOGY AND PROBLEM SOLVING**

Code: **CA1118**

Total Hours: **60 Hrs**

Hours/Week: **4**

Credits: **4**

Objectives

- To introduce the components of computers
- To introduce basic concepts of hardware and software.
- To introduce the general structure of the CPU, motherboard and advance interfaces
- To understand problem solving methodologies
- To introduce the elementary concepts of word processing, ESS and Web Technologies

Learning Outcome

- Understanding the fundamentals of computers.
- Understanding the functions of CPU and other components of the motherboard.
- Understanding the software concepts.

UNIT 1 **(12)**

FUNDAMENTALS OF COMPUTERS

Functional block diagram of a digital computer, Historical overview-history of computers, Generation of computers-up to the present, Application of computers, Classification of Computers- Analog, Digital, Hybrid, Micro, Mini, Mainframe computers etc.

INPUT, OUTPUT AND MEMORY DEVICES

Introduction to input devices: The Keyboard, The Mouse, The Joystick, The OMR, OCR, MICR. Introduction to output devices: The Monitors, The Printers, (Characteristics to be

Identified), The Speakers. Introduction to Memory devices: The Primary Memory, The Secondary Memory, The Cache Memory.

Self Study: Applications of Computers in new fields, Improved Input and Output devices.

UNIT 2 **(12)**

SOFTWARE CONCEPTS

Introduction, Types of Software (Application and System software), Introduction to computer languages (Machine level language, assembly level language, high level language), Language translators (Compilers, interpreters, linkers, loaders).

OPERATING SYSTEM CONCEPTS

Introduction to Operating Systems, Functions of Operating Systems, Types of Operating Systems, Functional features of commonly used operating systems

DATA PROCESSING CONCEPTS

Introduction to data processing, Difference between data and information, Examples of data processing systems, Data processing cycle and functions, Data Hierarchy, types of data processing systems (Manual and Electronic).

Self Study: Comparison of software, Applications of different languages. Understanding features of modern operating systems.

UNIT 3 **(12)**

COMPUTER COMPONENTS

General structure of the CPU, how instructions are executed, Factors affecting speed of a computer, Introduction to processors, Evolution of Microprocessors (Intel and AMD), Specifications of computer components (Motherboard), Power consumption of PC.

ADVANCED FEATURES

Advanced I/O interfaces, Configuring specifications of a computer, additional accessories such as UPS for various applications, Importance of computer maintenance, Computer viruses and Computer security.

Self Study: Comparison of computer components, describing the configuration of a computer for your college.

UNIT 4 **(12)**

PROBLEM SOLVING METHODOLOGY

Introduction to problem Solving, problem definition and problem Analysis, Design of a Solution- Algorithms and Flowcharts. Development of Programs (Coding, testing, debugging)-Documentation and Maintenance Programming Constructs- Sequence, Selection and Iteration Characteristics of a Good Program-Types of Errors, Approaches to Problem Solving-Top-down, Bottom-up, Modular, Structured and object oriented

Self Study: Apply problem solving techniques for different types of problems.

UNIT 5 **(12)**

WORD PROCESSING

Word Processing applications: creation of documents, Parts of the Menu/window, copy and move, formatting features, spell check, print, creation of tables and other basic operations.

SPREADSHEETS

Spreadsheet applications (elementary level), Basics concepts of spreadsheet and other features such as, entering text, menus, commands, column width, copy, paste, to insert rows/columns, formatting, formula, print, sort, filter and other basic operations. Some advanced features such as graphs, library functions (Arithmetic, Date and Time, Financial, Logical, text and statistical).

INTERNET CONCEPTS

Introduction to the Internet, Internet services, protocols.

TEXT BOOKS

- Fundamentals of Computers by Rajaraman, PHI, (6th Edition).
- Computer Organization by Carl Hamacher V. Zaki, McGraw Hill, (5th Edition).
- Digital computer Fundaments by Malvino& Leach, (6th Edition).
- Office Automation and word processing – E Balaguruswamy

REFERENCES

- Dummies Series – MS-Office (2016 Edition)
- M. Ercegovac, T. Lang, J.H. Moreno, **Introduction to Digital Systems**, John
- Wiley and Sons

BLUE PRINT OF THE QUESTION PAPER

UNITS	2 MARKS (10 Questions)	6 MARKS (7 Questions)	10 MARKS (3 Questions)	TOTAL MARKS
1	2	2	1	26
2	2	1	-	10
3	2	2	1	26
4	2	1	1	20
5	2	1	-	10
TOTAL	**10**	**7**	**3**	**92**

Title: **DISCRETE MATHEMATICS**

Code : **CA1218**

Total Hours : **60 Hrs**

Hours/Week: **4**

Credits: **4**

Objectives

- To inculcate in students the fundamental mathematical background in computer science.
- To know the abstraction, notation, and critical thinking in the Discrete Mathematics.

Learning outcome

- Construct mathematical arguments using logical connectives and quantifiers.
- Implementation of Graphs and Mathematical Models in the study of networks.
- To solve real-world problems using Discrete Mathematics.

UNIT 1 **(12)**

MATHEMATICAL LOGIC

Propositions , Logical connectives ,Compound propositions , Conditional and bi conditional propositions, Tautologies and contradictions ,Contra-positive , Logical equivalences and implications , De Morgan's Laws , Normal forms , Principal conjunctive and disjunctive normal forms , Rules of inference , Arguments , Validity of arguments.

UNIT 2 **(8)**

FUNCTIONS

Definitions of functions, Classification of functions, Types of functions, Examples, Composition of functions, Inverse functions, Characteristic function of a set, Permutation functions.

UNIT 3 **(12)**

SET THEORY

Basic concepts , Notations ,Subset , Algebra of sets , The power set , Ordered pairs and Cartesian product , Relations on sets ,Types of relations and their properties , Partial ordering , Poset , Hasse diagram , Lattices and their properties , Sub lattices .

UNIT 4 **(14)**

GROUPS

Groups , Properties , Subgroups , Cosets and Lagrange's theorem , Normal subgroups , Algebraic system with two binary operations , Preliminaries of Coding ,Hamming Metric ,Group codes: Basic notions of error correction , Error recovery in group codes.

UNIT 5 **(14)**

GRAPH THEORY

Definition of a Graph, Finite and Infinite Graphs, Incidence and Degree of a Vertex, Sub graphs, Walks, Paths, Circuits, Euler Graph and Hamiltonian Circuit.

TREES AND MATRIX REPRESENTATION

Properties of Trees, Distance and Centre in a Tree, Spanning Trees and Fundamental Circuits. Planar and Dual Graphs Planar Graphs, Kurtowski's two Graphs.

DIRECTED GRAPHS

Definition, Types of Directed graphs, Euler Digraphs, Trees with directed edges, and Fundamental Circuits in Digraphs. Digraphs, Adjacency Matrix of a Digraph.

TEXT BOOKS

- Trembly J.P and Manohar R, "Discrete Mathematical Structures with Applications to Computer Science", Tata McGraw–Hill Pub. Co. Ltd, New Delhi, 2003.
- Graph theory – NarasinghDeo

REFERENCES

- Ralph.P.Grimaldi, "Discrete and Combinatorial Mathematics: An Applied Introduction", Fifth Edition, Pearson Education Asia, Delhi, 2002.
- Bernard Kolman, Robert C. Busby, Sharan Cutler Ross, "Discrete Mathematical Structures", Fourth Indian reprint, Pearson Education Pvt Ltd., New Delhi.
- Kenneth H. Rosen, "Discrete Mathematics and its Applications", Fifth Edition, Tata McGraw – Hill Pub. Co. Ltd., New Delhi.
- Richard Johnsonbaugh, "Discrete Mathematics", Seventh Edition, Pearson Education Asia, New Delhi, 2002.

BLUE PRINT OF THE QUESTION PAPER

UNITS	2 MARKS (10 Questions)	6 MARKS (7 Questions)	10 MARKS (3 Questions)	TOTAL MARKS
UNIT 1	2	2	1	26
UNIT 2	2	1	-	10
UNIT 3	2	2	1	26
UNIT 4	2	1	1	20
UNIT 5	2	1	-	10
TOTAL	**10**	**7**	**3**	**92**

Title: **PROGRAMMING IN C**

Code : **CA1318**

Total Hours : **60 Hrs**

Hours/Week: **4**

Credits: **4**

Objectives

- The course provides students with a detailed study of programming techniques using C programming language.
- Good programming habits, proper logical thinking, algorithm and flowchart development, writing efficient programs are taught in the course.
- Detailed lab exercises covering all aspects of the language are prepared.
- To understand the programming paradims.

Learning Outcome

- To analyze problems efficiently and develop comprehensive logic to solve it.
- To write C programs using program constructs.

UNIT 1 **(12)**

INTRODUCTION TO PROGRAMMING CONCEPTS

Overview of C Language: History of C, Character set, C tokens, Identifiers, Keywords, Data types, Variables, Constants, Symbolic Constants , Operators in C, Hierarchy of Operators, Expressions, Type Conversions and Library Functions.

UNIT 2 **(12)**

MANAGING INPUT AND OUTPUT OPERATION

Managing Input and Output Operation: Formatted and Unformatted I/O Functions, Decision making, branching and looping: Decision Making Statements - if Statement, if–else statement, nesting of if-else statements, else–if ladder, switch statement, conditional operator, Looping - while, do-while, for loop, Nested loop, break, continue, and goto statements.

UNIT 3 **(12)**

ARRAYS

Declaring and Initializing, One Dimensional Arrays, Two Dimensional Arrays, Multi Dimensional Arrays - Passing arrays to functions. Strings: Declaring and Initializing strings, Operations on strings, Arrays of strings, passing strings to functions. Storage Classes - Automatic, External, Static and Register Variables.

UNIT 4 **(12)**

FUNCTIONS

Function Definition, prototyping, types of functions, passing arguments to functions, Nested Functions, Recursive functions.

UNIT 5 **(12)**

FILES AND POINTERS

File modes, File functions, and File operations, Text and Binary files, Command Line arguments. Structures-Declaring and Initializing, Nested structure, Array of Structure, Passing Structures to functions, Unions, typedef, enum, Bit fields. Declarations, Pointer arithmetic, Pointers and functions, Call by value, Call by reference, Pointers and Arrays, Arrays of Pointers, Pointers and Structures. static and dynamic memory allocation, Memory allocation functions.

Self Study: Data types, memory allocation and programs related to all the units.

TEXT BOOKS

- E. Balaguruswamy, "Programming In ANSI C", 6th edition, TMH Publications.
- Ashok N. Kamthane, "Programming with ANSI and Turbo C", Pearson Education.

REFERENCES

- Ashok N. Kamthaneet. al., "Computer Programming and IT", Pearson Education, 2011
- The C programming Language, Brain W Kernigham, Dennis M Ritchie, Prentice Hall Software Series.
- Mahapatra, "Thinking In C ", PHI Publications, 2014.
- YashwantKanetkar, "Let Us C", 15th Edition, PHP.

BLUE PRINT OF THE QUESTION PAPER

UNITS	2 MARKS (10 Questions)	6 MARKS (7 Questions)	10 MARKS (3 Questions)	TOTAL MARKS
UNIT 1	2	1	-	10
UNIT 2	2	2	-	16
UNIT 3	2	1	1	20
UNIT 4	2	1	1	20
UNIT 5	2	2	1	26
TOTAL	**10**	**7**	**3**	**92**

Title:	**DIGITAL FUNDAMENTALS AND LOGIC DESIGN**
Code:	**CA1418**
Total Hours :	**60 Hrs**
Hours/Week:	**4**
Credits:	**4**

Objectives

- To introduce basic postulates of Boolean algebra and shows the correlation between Boolean expressions
- To introduce the methods for simplifying Boolean expressions
- To introduce the formal procedures for the analysis and design of combinational circuits and sequential circuits
- To introduce the concept of computer memories

Learning Outcome

- Understand the digital number systems and logic circuits.
- Capability to solve logic function minimization.
- Differentiate between combinational and sequential circuits such as decoders, encoders, multiplexers, de-multiplexers, flip-flops, counters, registers.
- Understand the specifications of logic families and computer memories.

UNIT 1 **(14)**

INTRODUCTION TO NUMBER SYSTEM AND CODES

Number systems: Decimal numbers , Binary numbers : Counting in binary, The weighted structure of binary numbers, Octal numbers, hexadecimal numbers and their mutual conversions, Binary arithmetic : Addition, subtraction, multiplication and division of binary numbers, 1's and 2's complement, signed numbers, arithmetic operations(addition, subtraction) with signed numbers, 9's and 10's complement, BCD numbers, BCD addition , BCD subtraction, Gray code: Binary to Gray code conversion, Gray to Binary conversion, Weighted code : 8421 code and Non weighted codes : ASCII and EBCDIC

UNIT 2 **(14)**

BOOLEAN ALGEBRA AND LOGIC GATES

Boolean operations and expressions, Laws and rules of Boolean algebra, Demorgan's Theorem, Boolean expressions, Simplification of Boolean expression.

AND gate, OR gate, NOT gate, NAND gate , NOR gate , X-OR gate , X-NOR gate, The universal property of NAND gate and NOR gate, Realization of basic gates. Boolean expression for logic circuits, Karnaugh map (SOP with examples)

Self Learning - Universal property of NOR gate

UNIT 3 **(12)**

COMBINATIONAL LOGIC

Basic Adders : Half adder, Full adder, 4-bit Parallel adders, Subtractor : half subtractor , full subtractor (Implementation using logic gates), Decoders: 4 bit decoder, BCD to decimal decoder, Encoder : Decimal to BCD encoder, Multiplexer – 4 to 1 multiplexer, De-multiplexer: 1 to 4 demultiplexer .

UNIT 4 **(8)**

FLIP FLOPS

Latches : SR latch, Clocked flip-flops :SR flip-flop, D flip-flop, JK flip-flop, Positive edge triggered flip flops, Timing diagrams , Master slave JK flip-flop.

UNIT 5 **(12)**

REGISTERS AND COUNTERS

Modes of operation of registers (SISO, SIPO, PISO, and PIPO),Asynchronous counters (Four bit ripple counter, Decade counter), Synchronous counter (Four bit synchronous counter, Decade counter).

Self-Learning: Introduction to RAM (SRAM,DRAM) , ROM,PROM, EPROM, EEPROM

TEXT BOOKS

- Digital Computer Fundaments by Thomas C Bartee, McGraw Hill, VI Edition, Computer Organization by Carl Hamacher V. Zaki, McGraw Hill.

- Digital computer Fundaments by Malvino& Leach.
- Digital Computer Fundaments by Malvino

REFERENCES

.

- Malvino, **Digital Principles and Applications**, Tata McGraw Hill, 4th Edition
- M. Ercegovac, T. Lang, J.H. Moreno, **Introduction to Digital Systems**, John Wiley and Sons

BLUE PRINT OF THE QUESTION PAPE R

UNITS	2 MARKS (10 Questions)	6 MARKS (7 Questions)	10 MARKS (3 Questions)	TOTAL MARKS
UNIT 1	2	1	-	10
UNIT 2	2	2	-	16
UNIT 3	2	1	1	20
UNIT 4	2	1	1	20
UNIT 5	2	2	1	26
TOTAL	**20**	**42**	**30**	**92**

PRACTICALS

Title: **C PROGRAMMING LAB**
Code: **CA1P1**
Hrs / Week: **2 Hrs**
Credits: **1**

Objectives

- To learn problem solving techniques
- To teach the students to write programs in C and to solve the problems.

Learning Outcome

After the completion of this course the student would be able to

- Read, understand and trace the execution of programs written in C language.
- Implement programs with pointers and arrays, perform pointer arithmetic and use the pre-processor

PROGRAMS LIST

1. To demonstrate the usage of operators and data types in C

a. Write a program to print the size of all the data types with its modifiers supported by C and its range.

b. Write a program to calculate simple interest.

2. To demonstrate the usage of if, if-else, nested-if and switch

a. Write a program to find the largest, smallest and second largest of three numbers.

b. Write a program to accept marks of three subjects calculate the total percentage and output the result of the student.

c. Write a program to find the second largest of four numbers.

d. Write a program to calculate Julian date.

3. To demonstrate the usage of while, do-while and for loops

a. Write a program to find the sum of numbers from 1 to N.

b. Write a program to reverse a number.

c. Write a program to generate the Fibonacci series.

4. To demonstrate the concept of arrays and strings

a. Write a program to check whether a string is a Palindrome.

b. Write a program to insert a number at a given position in an array.

c. Write a program to arrange a list of numbers in ascending order.

d. Write a program to check whether a given matrix is symmetric or not.

e. Write a program to perform matrix multiplication.

5. To demonstrate the usage of functions and recursion

a. Write a program to check whether a given number is prime or not.

b. Write a program to find the roots of a quadratic equation

c. Write a recursive program to find the factorial of a number.

d. Write a recursive program to find x_y.

6. To demonstrate the concept of structures

a. Write a program to create a student structure and display the same.

7. To demonstrate the concept of pointers

a. Write a program using function to swap two numbers using pointers

Title: **OFFICE AUTOMATION AND DIGITAL LOGIC LAB**
Code: **CA1P2**
Hrs / Week: **2 Hrs**
Credits: **1**

Objectives

- To learn various tools of MS-Word application.
- To learn text formatting that will help the students in designing documentation and other official letters.
- To learn the important mathematical function in MS-Excel that helps in calculations.
- To learn all the layouts and designs of power point presentation that can help in presenting their activities, seminars and conferences.

Learning outcome

After the completion of this course the student would be able to

- Create various corresponding letter formats and MS-Excel spread sheets with various mathematical functions for statistical analysis and representations of various graphs.
- Create multimedia based power point presentation with both audio and video tools.

OFFICE AUTOMATION LAB LIST

1: Design a word document to personal letter.

2: Create company letter head.

3: Create simple news letter.

4: Create a memo:

5: Create a resume.

6: Create greeting card.

7: Create a cover page of a project report.

8: Create a mail merge letter.

9: Create a macro for inserting a picture and formatting the text.

10: Create a simple presentation to list simple dos commands, hardware, and software.

11: Create a worksheet with 4 columns, enter 10 records and find the sum of all columns.

12: Create a report containing the pay details of the employee.

13: Create a student result sheet.

14: Create a simple bar chart to high light the sales of a company for 3 different periods.

15: Create a pie chart for a sample data and give legends.

16: Create a worksheet importing data from database and calculate sum of all the columns.

17: Create a macro which creates a line chart using the data in the worksheet.

18: Create simple table for result processing.

19: Create query table for the result processing table.

20: Create a form to update/modify the result processing table.

21: Design a report to print the result sheet and marks card for the result.

DIGITAL LOGIC:

An activity that helps the students to develop a working models of any two sequential circuits and one combinational circuit with demonstration.

SECOND SEMESTER

Title: **DATA STRUCTURES USING C**

Code: **CA2118**

Total Hours : **60 Hrs**

Hours/Week: **4**

Credits: **4**

Objectives

Data Structure is considered as one of the fundamental paper towards a more comprehensive understanding of programming and application development. Student is expected to work towards a sound theoretical understanding of Data Structures and also compliment the same with hands on implementing experience.

- To learn the practical implementation of the data structures like stack, queue, array etc.
- To understand and implement different searching and sorting techniques.

Learning Outcome

- Understand the need for Data Structures when building Applications.
- Understand the need for optimized algorithm.
- Understanding the operations of insertion and deletion of a data.

UNIT 1 **(12)**

INTRODUCTION TO DATA STRUCTURES

Introduction. Classification of data structures. Operations on primitive and non-primitive data structures. Analysis of algorithms, space and time complexity, best and worst case, asymptotic notation, upper and lower bounds.

Self Study: Apply complexity calculation for different types of problems.

UNIT 2 **(12)**

LINEAR DATA STRUCTURES

Introduction, Stacks, Storage representation of stacks, operations on stacks, applications of stacks, queues, operations on queues, circular queues, operations on circular queues, applications of queues and circular queues, polish notation, translation of expression, evaluation of expression.

Self Study: Compare the applications of different data structures.

UNIT 3 **(12)**

LINKED LISTS

Introduction, Singly linked list, operations on linked lists, traversing a linked list, creating a linked list, adding nodes at various positions in a linked list, deletion of nodes, advantages and disadvantages, types of linked lists, applications of linked lists.

Self Study: Compare the applications of linked lists as a data structures.

UNIT 4 **(12)**

TREES

Introduction, terminologies and basic concepts, Binary tree, Storage representation of binary tree, tree traversals, Binary search tree, building a binary search tree, height balanced tree, AVL rotation.

Self Study: Compare the applications of trees as a data structures.

UNIT 5 **(12)**

SEARCHING AND SORTING

Sorting – Bubble sort , Insertion sort, selection sort, quick sort, merge sort.
Searching – Linear and Binary.

Self Study: Compare different searching and sorting methods.

TEXT BOOK

- Data Structure by Schaum Series(Revised 1ST Edition)

REFERENCES

- Fundamentals of Data Structure by Horowitz Sahni (2nd Edition).
- Data Structure by Dale and Lilly.
- S. Sahni, Data Structures, Algorithms and Applications, Tata McGraw Hill.
- Data Structures and Algorithm Analysis in C, Second Edition, Mark Allen Weiss, Pearson.

BLUE PRINT OF THE QUESTION PAPER

UNITS	**2 MARKS (10 Questions)**	**6 MARKS (7 Questions)**	**10 MARKS (3 Questions)**	**TOTAL MARKS**
UNIT 1	2	2	1	26
UNIT 2	2	1	-	10
UNIT 3	2	2	1	26
UNIT 4	2	1	1	20
UNIT 5	2	1	-	10
TOTAL	**10**	**7**	**3**	**92**

Title: **MICROPROCESSORS**

Code: **CA2218**

Total Hours: **60 Hrs**

Hours/Week: **4**

Credits: **4**

Objectives

- The course focuses on architecture of 8085 microprocessor and its operations.
- Understand how to write assembly program.
- The course also focuses on architecture of interfacing device 8255A.

Learning Outcome

- Identify the basic element, functions and architecture of 8085 microprocessor.
- Learn the instruction set of 8085 and code in the assembly language program to develop the microprocessor based application.

UNIT 1 **(12)**

ARCHITECTURE OF 8085 MPU

Introduction to 8085, Microprocessor architecture & its operation Microprocessor based system, memory interfacing , basic interfacing concepts ,interfacing I/O devices

UNIT 2 **(12)**

INTRODUCTION TO 8085 PROGRAMMING

Programming model, instruction classification , Instruction format, addressing modes, writing assembly level programs-overview of instruction set, timing diagrams data transfer, Arithmetic, Logic branch operations.

UNIT 3 **(12)**

PROGRAMMING TECHNIQUES WITH ADDITIONAL INSTRUCTIONS

Looping Counting and Indexing , 16 bit arithmetic operations , logic operations Compare and rotate operations . Counters and Time delays , Generation of pulse waveforms. Stacks and subroutines- conditional CALL and RETURN instructions. Advanced subroutine concepts. BCD to Binary and Binary to BCD conversions, BCD TO 7 segment conversion , Binary to ASCII and ASCII to Binary code conversion, BCD addition and subtraction , multiplication and division.

UNIT 4 **(12)**

INTERRUPTS

Implementing interrupts - Multiple interrupt - 8085 - trap - Problems on implementing 8085 interrupt - DMA - Memory interfaces - RAM and ROM - I/O interface - Direct I/O - Memory mapped I/O.

UNIT 5 **(12)**

8255APROGRAMMABLE PERIPHERAL INTERFACE

Block Diagram – Control Logic, Control Word – Modes of operations with examples, Mode 0, Mode 1, BSR Mode, Control word for each modes of operation Programming in 8255A with an example.

Self study : All instructions used in 8085 instruction set.

TEXT BOOK

- R. S. Gaonkar, 'Microprocessor Architecture, Programming and Applications with 8085/8080A'.

REFERENCE S

- A. Mathur, 'Introduction to Microprocessor', Third Edition, Tata McGraw-Hill Publishing Co. Ltd.
- 3.Hall, D.V. Microprocessor and Digital System, McGraw Hill Publishing Company, 2nd Edition, 2008.
- Charles M Gilmore & Pal, Ajit. Microprocessor Principles and Applications, Tata McGraw Hill, 2nd Edition.

BLUE PRINT OF THE QUESTION PAPER

UNITS	**2 MARKS (10 Questions)**	**6 MARKS (7 Questions)**	**10 MARKS (3 Questions)**	**TOTAL MARKS**
UNIT 1	2	2	1	26
UNIT 2	1	1	1	10
UNIT 3	3	2	-	26
UNIT 4	2	2	1	20
UNIT 5	2	1	-	10
TOTAL	20	42	30	**92**

Title: **OPERATING SYSTEM**

Code: **CA2318**

Total Hours: **60 Hrs**

Hours/Week: **4**

Credits: **4**

Objectives

- To introduce to the concept of Operating system.
- To acquire the fundamental knowledge of the operating system architecture and components
- To know the various operations performed by the operating system.

Learning Outcome

- Understand the basic working process of an operating system.
- Understand the importance of process and scheduling.
- Understand the problems in process synchronization and memory management.

UNIT 1 **(12)**

INTRODUCTION

Definition, functions, views, types, Buffering & spooling, multiprogramming, time - sharing, Real time system, protection, operating system structure, system components, system calls.

UNIT 2 **(12)**

PROCESS MANAGEMENT

Process concept(program, process and thread),states of a process, process control block, functions, Cooperating process, Inter-process communication, Scheduling Criteria, job and processor scheduling. (pre-emptive and non-preemptive) FCFS, SJF, Round Robin ,priority scheduling, multilevel, multilevel feedback algorithms.

UNIT 3 **(12)**

PROCESS SYNCHRONIZATION AND DEADLOCK

process synchronization – semaphores, critical section problems, classic problems of synchronization System model, deadlock characterization, methods for handling deadlock, deadlock prevention, avoidance and detection using resource allocation graph and wait for graph , Recovery from deadlock.

UNIT 4 **(12)**

MEMORY MANAGEMENT

Functions, Different schemes-Single continuous-Partitioned -Multiple Relocatable-Paging-Demand paging-Segmentation-Paged segmentation. Virtual Memory management: Demand paging, Page replacement and page replacement algorithm(FIFO,LRU), and thrashing.

UNIT 5 **(12)**

DEVICE AND FILE SYSTEM

Disk structure, allocation methods, free space management, need for disk scheduling, scheduling algorithm-FCFS (First Come, First Served), SSTF (Shortest Seek Time First) ,SCAN,C-SCAN (circular scan), access methods, directory structure, organization, file protections.

Self study: DOS- Internal and External commands

TEXT BOOK:

- Operating System by Milan Milenkovic, McGraw Hill.(5th Edition)

REFERENCES

- Operating System by Madnick and Donoval, McGraw Hill.
- Operating Systems: Internals and Design Principles, by William Stallings, seventh edition
- Operating System Concepts by James L Peterson. (2nd Edition)
- Operating System Design and Implementation by Andrew S Tenenbaum. (3rd Edition)

BLUE PRINT OF THE QUESTION PAPER

UNITS	**2 MARKS (10 Questions)**	**6 MARKS (7 Questions)**	**10 MARKS (3 Questions)**	**TOTAL MARKS**
UNIT 1	2	1	1	20
UNIT 2	2	2	1	26
UNIT 3	3	2	-	18
UNIT 4	2	1	1	20
UNIT 5	1	1	-	08
TOTAL	20	42	30	**92**

Title:	**NUMERICAL METHODS AND STATISTICS**
Code:	**CA2418**
Total Hours:	**60 Hrs**
Hours/Week:	**4**
Credits:	**4**

Objectives

- To enable the students to obtain an intuitive and working understanding of numerical methods for the basic problems of numerical analysis and gain an experience in the implementation of numerical methods using a computer.
- To learn the concept of error in Numerical Analysis and predict it.
- To know the statistical tools in analyzing the various data.

Learning Outcome

- To be aware of the use of numerical methods in modern scientific computing
- To be familiar with finite precision computation
- To be familiar with calculation and interpretation of errors in numerical methods
- Understand the basic concepts of statistical methods
- Develop analytical ability to solve real-world problems using these methodologies

UNIT 1 **(10)**

COMPUTER ARITHMETIC AND SOLUTION OF NON-LINEAR EQUATIONS

Introduction – Floating Point Arithmetic and Errors: Floating point representation of Numbers, Arithmetic operations with Normalized Floating Point Numbers, Consequences of Normalized Floating Point Representation of numbers. Pitfalls in Computation. Errors in numbers.

UNIT 2 **(12)**

SOLUTION OF NON-LINEAR EQUATIONS

Iterative method, Successive Bisection– Fixed point – Regulafalsi – Newton's Raphson – Secant method. Solution of simultaneous Linear Algebraic Equations and ordinary differential equations Cramer's Rule. Gauss Elimination method – Pivoting Strategies .Gauss Seidal Iterative Method.

UNIT 3 **(12)**

INTERPOLATION AND CURVE FITTING

Problem of Interpolation – Lagrange's method of Interpolation – Inverse Interpolation – Newton's interpolation formulae – Error of the Interpolating Polynomial - Interpolation at equally spaced points : Forward and Backward differences – Newton's forward and backward difference formulas. Fitting of polynomials and other curve - Least square approximation of functions - linear and polynomial regressions.

UNIT 4 **(14)**

INTRODUCTION TO STATISTICS

Overview of Statistics – Basic concepts – Population, Sample, Types of data - Primary and Secondary data, qualitative, quantitative, cross sectional, time series, variables and attributes, discrete and continuous variables, types of scales - nominal, ordinal, ratio and interval, frequency distribution, construction of tables with one or more factors of classification and graphical and diagrammatic presentation of various types of data, central tendency and their measures, dispersion and their measures and introduction to skewness and Kurtosis, random experiment, trial, sample space, events, classical, empirical and axiomatic approaches to probability, properties of probability, additive law, conditional probability, multiplicative law and their applications, random variable,

UNIT 5 **(12)**

ESTIMATION AND REGRESSION

Discrete and continuous probability distribution with some examples, distribution function, mathematical expectation and properties, Bernoulli, Binomial, Geometric, Poisson, Uniform, Exponential, normal and chi-square distributions – with examples.

Point Estimation - Concepts of parameter, statistic, estimator and estimate, criteria for good estimator unbiasedness and consistency, methods of estimation, maximum likelihood estimator (MLE), concept of Interval estimation, definition of statistical hypotheses and types - null and alternative.

TEXT BOOKS

- Rajaraman,"Computer Oriented Numerical Methods",Prentice-Hall of India Pvt Limited.
- Gupta, S.C., and V.K.Kapoor : Fundamentals of Mathematical Statistics: Sultan Chand & Sons

REFERENCES

- P.Thangaraj,"Computer Oriented Numerical Methods",Prentice-Hall of India Pvt Limited.
- N.Datta,"Computer Oriented Numerical Methods",Vikas Publishing House Pvt Limited.
- S.S.Sastry "Introductory methods of Numerical Analysis", Prentice -Hall of India Pvt Limited.

BLUE PRINT OF THE QUESTION PAPER

UNITS	**2 MARKS (10 Questions)**	**6 MARKS (7 Questions)**	**10 MARKS (3 Questions)**	**TOTAL MARKS**
UNIT 1	2	2	1	26
UNIT 2	2	1	1	20
UNIT 3	2	1	-	10
UNIT 4	2	2	1	26
UNIT 5	2	1	-	10
TOTAL	20	42	30	**92**

PRACTICALS

Title:	**DATA STRUCTURES LAB**
Code:	**CA2P1**
Hrs / Week:	**2 Hrs**
Credits:	**1**

Objectives

- Understanding basic data structures and algorithms.
- Formulate new solutions for programming problems that improves existing code using algorithms and data structures.
- To evaluate algorithms and data structures in terms of time and memory complexity of basic operations.

Learning Outcome

- Student would be able to know the efficiency of an algorithms and its applications in solving real time problems.

PROGRAMS LIST

1. Call by value and call by reference
2. Insertion Sort
3. Selection Sort
4. Quick Sort
5. Bubble Sort
6. Linear Search
7. Binary Search
8. Length of a string using pointer
9. Concatenate two strings using pointers
10. Copy a string using pointers
11. Array implementation of a stack.
12. Array implementation of a queue
13. Array implementation of circular queue.
14. Creating a linked list.
15. Adding nodes at various positions in a linked list.
16. Deleting nodes from various positions from a linked list.
17. Conversion of infix expression to postfix expression.
18. Evaluation of postfix expression
19. Creating a binary search tree and performing the various traversals on a binary search tree.

Title: **MICROPROCESSOR LAB**
Code: **CA2P2**
Hrs / Week: **2 Hrs**
Credits: **1**

Objectives

- To understand the basic instructions of 8085.
- To write Programs and execute with the help of 8085 simulator

Learning Outcome

- Student would be able to know the applications of 8085.

PROGRAMS LIST

1. Exchange of two 16-bit numbers.
2. Addition and Subtraction of two 8-bit HEX numbers.
3. Addition of two 16-bit numbers.
4. Subtraction of two 16-bit numbers.
5. Conversion of hexadecimal to BCD.
6. Block Transfer.
7. Block Transfer in reverse order.
8. 2's Complement of 8 and 16 bit number.
9. Largest and smallest number in an Array
10. Checking of positive, negative and Zero.
11. Multiplication by successive addition.
12. Frequency of occurrence of a digit in a number.
13. Sum and Average of n-one byte number
14. Square root of a given number.
15. Searching for a number from a list.
16. Addition of two 32 bit numbers.
17. Counting number of ones and zeroes.
18. Sum of odd and even numbers in an array.
19. Add “N” one byte numbers.
20. Check whether a number is palindrome or not.

Course Structure and Syllabus for

Bachelor of Computer Applications (BCA)

(Six Semester Course)

Proposed for 2018 batch onwards

With

CHOICE BASED CREDIT SYSTEM (CBCS)

Department of Computer Science
Faculty of Science
St. Joseph's College (Autonomous)

DEPARTMENT OVERVIEW

VISION STATEMENT

The vision of the department is to make possible for our students to have a deep and analytical understanding of the field and to enable them to use their immense potential to enhance the quality of human life.

MISSION STATEMENT

The mission of the department is to offer a high-quality science education in the science of computing, as well as to prepare students for career opportunities in this area requiring a high level of technical knowledge and skill. The department will continue to develop and administer programs which serve the needs of industrial computer scientists, researchers, and computer application specialists for whom the discipline of computing sciences provides indispensable tools.

INTRODUCTION TO PROGRAM

Bachelor of Computer Applications is a three year six semester undergraduate program . The course is designed to function as an intermediate between the industry and academic institutes. The curriculum includes the latest technologies to prepare the student for the future. The student gains a strong foundation and skills in the field. The infrastructure provides an excellent environment for the student to contribute effectively in the field.

PROGRAM OBJECTIVE

Provide a strong foundation in fundamentals of computers. Prepare the students with exceptional skills of problem solving, communication and leadership skills. Facilitate overall understanding of the requirements of the subjects. Prepare the students to provide professional solutions to real time problems.

THIRD SEMESTER

Theory							
Sl.No.	**Code**	**Course Title**	**Hrs / Week**	**Credits**	**IA**	**SE**	**Total**
1	-	English	4	4	30	70	100
2	-	Language	4	4	30	70	100
3	CA3118	Object Oriented Programming Using Python	4	4	30	70	100
4	CA3218	.NET Technologies	4	4	30	70	100
5	CA3318	Software Engineering	4	4	30	70	100
6	CA3418	Database Management System	4	4	30	70	100
Practical							
1	CA3P1	Python Programming Lab	2	1	70	30	100
2	CA3P2	.NET and Database Management Lab	2	1	70	30	100

FOURTH SEMESTER

Theory							
Sl.No.	**Code**	**Course Title**	**Hrs / Week**	**Credits**	**IA**	**SE**	**Total**
1	-	English	4	4	30	70	100
2	-	Language	4	4	30	70	100
3	CA4118	JAVA Programming	4	4	30	70	100
4	CA4218	UNIX Programming	4	4	30	70	100
5	CA4318	Data Communication and Computer Networks	4	4	30	70	100
Open Electives (For students of other courses)							
6	CAOE4418	Web Development	4	2	15	35	50
Practical							
1	CA4P1	Java Programming Lab	2	1	70	30	100
2	CA4P2	UNIX Programming Lab	2	1	70	30	100

EXAMINATION AND ASSESMENTS

1.	**IA Weight age**	**30 %**
2.	**End Semester Examination Weight age**	**70%**

QUESTION PAPER PATTERN

The question papers of the theory examinations should follow the pattern specified below:

Section	**Marks for each question**	**Number Of Questions**		**Total Marks**
		Total	**Should Answer**	
A	2	10	10	20
B	6	7	5	30
C	10	3	2	20

Total Marks **70**

While selecting the questions importance should be given to all major units.

PRACTICAL QUESTION PAPER FORMAT

Scheme of valuation:

1.	Writing two programs one from each section	25 marks
2.	Execution of one program	25 marks
3.	Formatting the program and output	10 marks
3.	Record verification	05 marks
4.	Viva voce related to practical topics only	05 marks
	Total	70 marks

PROJECT EVALUATION FORMAT

Scheme of valuation:

1.	Demonstration and presentation	35 marks
2.	Documentation	35 marks
	Total	70 marks

INTERNAL ASSESSMENT FORMAT

THEORY:

1.	IA test	15 marks
2.	First Activity	10 marks
3.	Attendance	5 marks
	Total	30 marks

PRACTICALS:

Every practical class the student should be assessed.

1.	Writing the observation book	3 marks
2.	Executing the programs	5 marks
3.	Record writing	2 marks
	Total	10 marks

Internal marks for the final semester project work can be awarded by the guide by evaluating the performance of the student during the course of the project work.

THIRD SEMESTER

TITLE : OBJECT ORIENTED PROGRAMMING USING PYTHON

CODE : CA3118

Hrs / Week : 4 Hrs

CREDITS 4

Course Objective

To familiarize with Object Oriented concepts
To develop the skills of programming in Python

Learning Outcome

An understanding of the principles behind the object oriented development process.
Improve Programming skills
Hands on Regular Expressions and Files

UNIT I: (12)

INTRODUCTION TO PYTHON

Getting Started with Python, Essentials of Python Programming, Integers, Floating point Numbers, Strings, Variables and its Scope: Variables, Modifying values, Type Conversion, Selection Control Statements, Collections: Lists, Tuples, Sets, Ranges, Dictionaries, Two Dimensional Sequences, Loop Control Statements.

UNIT II: (12)

INTRODUCTION TO OBJECT ORIENTED PROGRAMMING IN PYTHON

Programming Paradigms, Objects, Classes, Python Classes, Specifying Attributes and Behaviors, Abstraction, Encapsulation , Polymorphism, Inheritance, Composition.

UNIT III: (12)

PROGRAMMING IN PYTHON

Creating Python Classes- Adding Attributes and making it perform a task, Initializing the Object, Modules and Packages, The Object-Oriented Approach, Custom Classes: Attributes and Methods, Inheritance and Polymorphism, Functions and Constructors , Using Properties to Control Attribute Access, Creating Classes That Aggregate Collections
,Creating Collection Classes Using Aggregation , Creating Collection Classes Using Inheritance Operators, Built-in Functions, List Type Built-in Methods, Special Features of Lists, Tuples, Tuple Operators .

UNIT IV: **(12)**

DESIGN PATTERNS , EXCEPTION HANDLING AND GUI

Raising an Exception, Catching and Raising Exceptions, Handling Exceptions, The Exception Hierarchy, Defining our Own Exceptions , Introduction to GUI Programming ,Dialog-Style Programs, Main-Window-Style Programs: Creating a Main Window

UNIT V: **(12)**

REGULAR EXPRESSIONS

Python‘s Regular Expression Language : Characters and Character Classes, Quantifiers, Grouping and Capturing, Assertions and Flags, The Regular Expression Module.

TEXT BOOKS

- Chun, J Wesley, Core Python Programming, Second Edition, Pearson, 2007 Reprint 2010
- Barry, Paul, Head First Python, 2nd Edition, O Rielly, 2010
- Lutz, Mark, Learning Python, 4th Edition, O Rielly, 2009
- Dusky Phillips, Python 3 Object-oriented Programming - Second Edition

BLUE PRINT OF THE QUESTION PAPER

UNITS	2Marks (10 Questions	6Marks (7 Questions)	10 Marks (3 Questions)	Total Marks
UNIT 1	2	1	1/2	15
UNIT 2	2	1		10
UNIT 3	2	2	1	26
UNIT 4	2	2	1/2	21
UNIT 5	2	1	1	20
TOTAL	20	42	30	92

TITLE : .NET TECHNOLOGIES

CODE : CA3218

Hrs / Week : 4 Hrs

CREDITS 4

Course Objective:

This course introduces differences between the procedural languages and event-driven languages.
Define and modify the properties and methods associated with an object.
Load, modify, and save changes made to forms and projects in the .NET environment.
Identify and perform the steps necessary to convert .NET programs to executable files that will run in the Windows environment.

Learning Outcomes:

On successful completion of this course, students should be able to:

To design and develop Windows-based business applications using .NET programs that meet commercial programming standards.
Work equivalent to that expected from someone already working in the information technology field as a professional programmer.
Grade you according to commercial standards.

UNIT I: (12)

FEATURES OF C# AND OOP

Operators in C#, expression, Decision making and looping statements in c# Methods in C# -declaring methods, nesting of methods, methods parameters, the output parameters, Static data members, Static member function, Arrays - variable size arrays, array list class, String handling . Overview of OOP , Defining a class, Adding Variables and Methods, Member Access Modifiers, Creating Objects, Accessing Class Members,

UNIT II: (12)

INTRODUCTION

.NET Definition, Characteristics, Application, Difference between c and c#, The .NET Strategy, The origin .NET framework, benefits of the .NET approach, C# and .NET, .NET Architecture-Common Runtime, MSIL, .NET Framework Classes. Advantages of Managed Code ,Garbage Collection , Security. program structure , command line argument, Literals, variables and data types , constant variables, scope of variables, boxing and unboxing.

UNIT III: **(12)**

OBJECT ORIENTED C#

Types of Constructors ,This Reference, Inheritance and polymorphism :classical inheritance, containment inheritance, defining a subclass, visibility control, defining subclass constructors, multilevel inheritance, hierarchical inheritance, overriding methods, hiding methods, abstract classes, abstract methods, sealed classes : Preventing inheritance, sealed methods, polymorphism. Interfaces : Multiple Inheritance : defining an interface, extending an interface, implementing interface, interface & inheritance, explicit interface implementation.

UNIT IV: **(12)**

INTRODUCTION TO VISUAL STUDIO AND EXCEPTIONS

NET- ASP.NET. Difference Between ASP and ASP.NET, Creating A web application using ASP.NET ,Components of an ASP.NET Custom Control, Deploying ASP.NET application .Errors and Exception Handling, Exceptions Classes User Defined Exceptions , Delegate- delegate declaration, delegate methods, delegates instantiation, delegate invocation, using delegates.

UNIT V: **(12)**

INTRODUCTION TO DATA ACCESS

Comparison between ADO & ADO.NET—The difference between Connection Model & Disconnected Model – difference between the DataSet and RecordSet- The Dataset Model. Accessing Data using ADO.NET: dataset- DataAdapterDataRelation. The two Managed Providers: SQL Managed Provider-OLEDB Managed Provider. The ADO.NET Object Model: OLE DB Connection /SQLConnectionOleDbCommand/SQLCommand,OleDbDataReader/SQLDataReaderOleDbD ataAdapter/SQLDataAdapter-The Data Set. Using the Binding Manager to bind controls to the data - Working with Master-Detail relationship.

Self Study: Master Detail Relationships

TEXT BOOKS:

- Simon Robinson, Christian Nagel, Karli Watson, Jay Glynn, Morgan Skinner and Bill Evjen, Professional C#, Wiley – dreamtech India Pvt. Ltd., 3rd Edition, 2004.
- .NET(Core Reference) Microsoft® Visual C#® 2005: The Language by Donis Marshall
- Programming in C#, E. Balagurusamy, 2016, McGraw Hill Education
- Beginning ASP.NET in C#, Matthew MacDonald
- Designing Microsoft ASP.NET Application , Jonathan Goodyear, Brain Peek

BLUE PRINT OF THE QUESTION PAPER

UNITS	2 MARKS (10 Questions)	6 MARKS (7 Questions)	10 MARKS (3 Questions)	TOTAL MARKS
UNIT 1	2	1	-	10
UNIT 2	2	1	1/2	15
UNIT 3	2	2	1/2	21
UNIT 4	2	1	1	20
UNIT 5	2	2	1	26
TOTAL	20	42	30	92

TITLE : SOFTWARE ENGINEERING

CODE : CA3318

Hrs / Week : 4 Hrs

CREDITS 4

Course Objective

To inculcate in students different concepts of software engineering principles
To develop the skills necessary to design, develop and execute software projects.
Learning Outcome On completion of the course the student will:
Understand the importance of the stages in the software life cycle.
Understand the various process models.
Be able to design software by applying the software engineering principles.

Learning outcome

Upon successful completion of this course student should be
able to Understand the importance of software life cycle.
Understand the various process models.
Design and develop software by applying the software engineering principles.
Implement developed software efficiently and effectively.

UNIT I : (12)

SOFTWARE PROCESS AND SOFTWARE PROCESS MODELS

Software process, project and product, process assessment, Software Process capability maturity model: CMM Model. Life cycle models: Waterfall model, Incremental model, spiral model, Prototyping Model, Agile model, Extreme programming, Scrum models

UNIT II: (12)

SOFTWARE REQUIREMENTS AND DESIGN

Functional- non-functional requirements, User requirement, System requirements, Software requirements documentation. Software Requirement engineering ,process Feasibility studies, Requirements elicitation and analysis, requirement validation, software prototyping. Software Design: Design Process ,Design strategies ,Design Quality, System Structuring, control models, Modular decomposition, Domain Specific architecture.

Self Study: Data Flow Diagrams

UNIT III: (12)

SOFTWARE CODING AND IMPLEMENTATION

Implementation, Structures coding techniques, coding styles, Coding methodology, Coding verification techniques, Coding tools, code documentation, Software maintenance, Software re-engineering, Change management, configuration management.

UNIT IV: **(12)**

SOFTWARE TESTING AND MAINTENANCE

testing strategies, A strategic approach to software testing, test strategies for convention software, Black-box and white box testing, validation and system testing, clean room software development. Project Management, Quality Management
,Software cost estimation, software maintenance

UNIT V: **(12)**

QUALITY AND PROJECT MANAGEMENT

The management spectrum, Metrics in the process and project domains, software measurement, metrics for software quality, The project planning process, Software project estimation, Decomposition techniques, Empirical estimation models, Project scheduling, Risk identification, Risk projection. Software quality, Achieving software quality, Review metrics and their use, Informal reviews, Formal technical reviews.

TEXT BOOKS:

- Roger S. Pressman, "Software Engineering: A Practitioner's Approach", Alternate Edition, 9th
- Edition, Mcgraw Hill,
  2018.
- Ian Sommerville, "Software Engineering", 8th Edition, Pearson Publication Ltd 2012 .
- CarloGhejgietal, Fundamentals Of Software Engineering, Pearson Education.
- Panakajjalote, An Integrated Approach To Software Engineering – Narosa Publishing House

BLUE PRINT FOR QUESTION PAPER

UNITS	2 MARKS (10 Questions)	6 MARKS (7 Questions)	10 MARKS (3 Questions)	TOTAL MARKS
UNIT 1	2	1	1	20
UNIT 2	2	1	-	10
UNIT 3	2	2	1	26
UNIT 4	2	1	-	10
UNIT 5	2	2	1	26
TOTAL	20	42	30	92

TITLE : DATABASE MANAGEMENT SYSTEMS

CODE : CA3418

Hrs / Week : 4 Hrs

CREDITS 4

Course Objective:

To provide strong foundation of database concepts and develop skills for the design and implementation of a database application with a brief exposure to advanced database concepts.

Learning Outcome:

Understanding the core terms, concepts, and tools of relational database management systems. Understanding database design and logic development for database programming.

UNIT I: (12)

INTRODUCTION TO DBMS AND DBMS ARCHITECTURE

Data, Database, Database management system, Characteristics of the database approach, Role of Database administrators, Role of Database Designers, End Users, Advantages of Using a DBMS and When not to use a DBMS. DBMS Architecture, Data Models – Categories of data models, Schemas, Instances, and Database state. DBMS Architecture and Data Independence – The Three schema architecture, Data independence. DBMS Languages and Interfaces. Classifications of Database Management Systems.

UNIT II: (12)

DATA MODELING USING ER MODEL , UNDERSTANDING FILES

Using High Level Conceptual Data Models for Database Design, Example Database applications. Entity types, Entity Sets, Attributes and Keys. Relationships, Relationship types, Roles and Structural constraints, Drawing E- R Diagrams. Single Level Ordered Indexes – Primary indexes, Clustering indexes and Secondary indexes. Multi-level indexes, Dynamic Multilevel indexes using B-trees Introductory concepts and hashing.

Self Study: E-R Diagrams, Drawing E- R Diagrams.

UNIT III: (12)

RELATIONAL DATA MODEL AND DATABASE DESIGN

Relation, Integrity constraints - domain, entity and Referential integrity constraints, Basic Relational Algebra operations, select, project and join operations. Functional dependencies and Normalization for Relational Databases Normalization concepts, first, second, third normal forms, Boyce-Codd normal form.

UNIT IV: **(12)**

SQL, TRANSACTION AND CONCURRENCY CONTROL TECHNIQUES

SQL data definition and data types, specifying constraints in SQL, schema change statements, Basic queries, More Complex SQL queries, INSERT, DELETE and UPDATE statements in SQL, Views – Concept of a view in SQL. Transaction Processing Concepts and Concurrency Control Techniques, Desirable properties of Transactions – Schedules and Recoverability. Lock-Based Protocols – Locks, Granting of Locks, and Two phase locking protocol and implementation of locking.

UNIT V: **(12)**

DISTRIBUTED DATABASE AND DATABASE RECOVERY

Distributed Databases-Distributed database concepts, Data fragmentation, Replication, and Allocation Techniques for Distributed database design, Types of Distributed database systems. Database Recovery- Recovery Concepts: Recovery Outline and Categorization of Recovery Algorithms

Text Books :

- Elmasri & Navathe, Fundamentals of Database Systems , Pearson Education.
- Karate, Introduction to Database Management System, Pearson Education.
- Abraham. Silberschatz, Henry. F. Korth, S. Sudarshan, Database System Concepts, McGraw hill.
- Jeffrey. D. Ullman, Principles of database system.
- Oracle Press: ORACLE- Computer reference
- C.J. Date, Introduction to database systems, Eighth Edition.
- Data base System Concepts, Silberschatz, Korth, McGraw hill, Sixth Edition.
- Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, TATA McGrawHill 3rd Edition.
- An Introduction to Database systems, C.J. Date, A.Kannan, S.Swami Nadhan, Pearson, Eight

BLUE PRINT FOR QUESTION PAPER

UNITS	2 MARKS (10 Questions)	6 MARKS (7 Questions)	10 MARKS (3 Questions)	TOTAL MARKS
UNIT 1	2	1	1	20
UNIT 2	2	1	-	10
UNIT 3	2	2	1	26
UNIT 4	2	1	-	10
UNIT 5	2	2	1	26
TOTAL	20	42	30	92

PRACTICALS

TITLE : PYTHON PROGRAMMING LAB

CODE : CA3P1

Hrs / Week : 2 Hrs

CREDITS 1

THIRD SEMESTER

TITLE: PYTHON PROGRAMMING LAB
PAPER CODE: CA3P1
CREDIT: 1

1. Demonstrate running python programs in interactive interpreter.
2. Demonstrate running python program in IDLE.
3. Demonstrate running python program in command prompt.
4. Write a program that defines a matrix to perform addition of matrix
5. Program to print largest Even and Largest odd number in a List.
6. Program to show the usage of Tuples.
7. Program to count the occurrences of each word in a given string sentence.
8. Program to accept three digits and print all possible combination of the digits
9. Program to print sum of negative number ,positive number ,even number and odd number in a list.
10. Progam to create a class which performs basic calculator operations.
11. Program to create a class and get all possible subsets of a set.
12. Program to demonstrate constructor and destructor in python.
13. Demonstrate a program to implement inheritance.
14. Demonstrate a program to implement polymorphism.
15. Demonstrate a program to implement operator overloading.
16. Demonstrate a program to implement exception handling
17. Demonstrate a program to implement package.
18. Demonstrate a program to implement GUI
19. Demonstrate a program to implement Regular expression.
20. Demonstrate a program to implement quantifiers and grouping

TITLE : DATABASE and .NET LAB

CODE : CA3P2

Hrs / Week : 2 Hrs

CREDITS 1

DBMS and .NET Lab Programs

PROGRAM LIST

1. Write a c# program to demonstrate the working of command line arguments.
2. Write a c# program to demonstrate the working of out method and parameter array.
3. Write a c# program to demonstrate the working of jagged array.
4. Write a c# program to demonstrate the working of mutable and immutable string.
5. Write a c# program to demonstrate the working of delegates.
6. Write a c# program to demonstrate the working of multi level inheritance.
7. Write a c# program to demonstrate the working of multiple interfaces.
8. Write a c# program to demonstrate the working of virtual and override keyword.
9. Write a c# program to demonstrate the working of sealed class.
10. Write a c# program to demonstrate the working of abstract class and methods.
11. Write a c# program to demonstrate the working of indexer.
12. Write a c# program to demonstrate the working of exception handling.
13. Write a c# program to demonstrate the working of multithreading.
14. Write an asp.net program to demonstrate the working of simple calculator.
15. Database connectivity using ADO.net

FOURTH SEMESTER

TITLE : JAVA PROGRAMMING

CODE : CA4118

Hrs / Week : 4 Hrs

CREDITS 4

Course Objective:

This course introduces fundamental structured and object-oriented programming concepts and techniques, using Java, and is intended for all who plan to use computer programming in their studies and careers. Topics covered include variables, arithmetic operators, control structures, arrays, functions, recursion, dynamic memory allocation, files, class usage , arrays, recursion, polymorphism, exceptions ,Applet Programming and class writing. Program design and testing are also covered, in addition to more advanced object-oriented concepts including inheritance and exceptions as time permits.

Learning Outcomes:

On successful completion if this course, students should be able to:

Understand the basic concepts and principles of structured programming.
Understand the basic concepts and principles of object oriented programming.
Design, write, and test a Java program to implement a solution to a given problem Specification.

UNIT I: **(12)**

INTRODUCTION TO JAVA

Introduction to Programming Languages, Evolution of Java , Object- Oriented Programming Concepts and Java, Differences between C++ and Java, The Primary Characteristics of Java, The Architecture, Simple Java Program **,**Classes, Arrays, Strings. Constructors, Methods Overloading, Static Members, Nesting of Methods. Inheritance: Extending a Class Overriding Methods, Final Variables and Methods, Finalizer methods, Abstract Methods and Classes, Visibility Control. Arrays Strings and Vectors: Arrays, One – dimensional Arrays, Creating an Array, Two – dimensional Arrays, Strings, Vectors, Wrapper Classes.

UNIT II: **(12)**

INTERFACES AND PACKAGES

Interfaces- Multiple Inheritance: Introduction, Defining Interfaces, Extending Interfaces, Implementing Interfaces, Accessing Interface Variables. Packages: Putting Classes together: Introduction, Java API Packages, Using System Packages, Naming Conventions, Creating Packages, Accessing a Package, Using a Package, Adding a Class to a Package, Hiding Classes.

UNIT III: **(12)**

MULTITHREADING AND EXCEPTION HANDLING

Multithreaded Programming**-** Introduction, Creating Threads, Extending the Thread Class, Stopping and Blocking a thread, Life Cycle of a thread, Using Thread Methods, Thread Exceptions, Thread Priority, Synchronization, Implementing the'Runnable' Interface.

MANAGING EXCEPTIONS

Introduction, Types of Errors, Syntax of Exception Handling Code, Multiple Catch Statements, Using Finally Statement, Throwing Our Own Exceptions.

UNIT IV: **(12)**

MANAGING FILES AND APPLETS

Managing Input / Output Files in JAVA**-** Introduction, Concept of Streams, Stream Classes, Byte Stream Classes, Character Stream Classes.

APPLET PROGRAMMING

Introduction, How Applets Differ from Applications, Preparing to Write Applets, Building Applet Code, Applet Life Cycle, Creating an Executable applet, Designing a Web Page, Applet Tag, Adding Applet to HTML File, running the Applet.

Self Study: Managing Files

UNIT V: **(12)**

EVENT HANDLING AND JDBC

Event handling mechanisms-Delegation Event Model-Event classes-Sources of events-Event listener interfaces-Handling mouse and keyboard events- Adapter classes-Inner classes. Introduction to JDBC,JDBC Drivers, JDBC Architecture.

TEXT BOOKS:

- Schidt Herbert ,Java: The Complete Reference, Tata McGraw-Hill,8th Edition,2011
- E.Balaguruswamy,Programming with JAVA a primer ,Tata McGraw-Hill publishing Company Limited ,Delhi,4th Edition, 2010

BLUE PRINT FOR QUESTION PAPER

UNITS	2 MARKS (10 Questions)	6 MARKS (7 Questions)	10 MARKS (3 Questions)	TOTAL MARKS
UNIT 1	2	1	1	20
UNIT 2	2	1	-	10
UNIT 3	2	2	1	26
UNIT 4	2	1	-	10
UNIT 5	2	2	1	26
TOTAL	20	42	30	92

TITLE : UNIX PROGRAMMING

CODE : CA4218

Hrs / Week : 4 Hrs

CREDITS 4

Course Objectives:

This course will enable students to:

Understand the UNIX Architecture, File systems and use of basic Commands.
Use of editors and Networking commands.
Understand Shell Programming and to write shell scripts.
Understand and analyze UNIX System calls, Process Creation, Control & Relationship.

Course outcomes:

After studying this course, students will be able to:

Explain multi user OS UNIX and its basic features
Interpret UNIX Commands, Shell basics, and shell environments
Design and develop shell programming, communication, System calls and terminology.
Design and develop UNIX File I/O and UNIX Processes.

UNIT I **(12)**

INTRODUCTION

Introduction, Brief history. Unix Components/Architecture. Features of Unix.The UNIX Environment and UNIX Structure, Posix and Single Unix specification. The login prompt. General features of Unix commands/ command structure. Understanding of some basic commands such as echo, printf, ls, who, date, passwd, cal. Meaning of Internal and external commands. The type command: knowing the type of a command and locating it. The man command, more command ,Knowing the user terminal. The root login. Becoming the super user: su command. The /etc/passwd and /etc/shadow files. Commands to add, modify and delete users.

UNIT II **(12)**

UNIX FILES

Naming files. Basic file types/categories. Parent child relationship. The home directory and the HOME variable. Reaching required files- the PATH variable, manipulating the PATH, Relative and absolute pathnames. Directory commands – pwd, cd, mkdir, rmdir commands. The dot (.) and double dots (..) notations to represent present and parent directories and their usage in relative path names. File related commands – cat, mv, rm, cp, wc and od commands. File attributes and permissions. The ls command with options. Changing file permissions.

UNIT III **(12)**

THE VI EDITOR

Basics. Different ways of invoking and quitting vi. Different modes of vi. Input mode commands. Command mode commands. The ex mode commands. Illustrative examples Navigation commands. Repeat command. Pattern searching. The search and replace command. Simple examples using these commands. The shells interpretive cycle. Wild cards and file name generation. Three standard files and redirection. Connecting commands: Pipe. Splitting the output: tee. Command substitution. Basic and Extended regular expressions. The grep. Typical examples involving different regular expressions.

UNIT IV **(12)**

SHELL PROGRAMMING

Shell programming. Ordinary and environment variables. The .profile. Read and read only commands. Command line arguments. exit and exit status of a command. Logical operators for conditional execution. The if, while, for and case control statements. The set and shift commands and handling positional parameters. Simple shell program examples.

UNIT V **(12)**

FILE INODS AND PROCESS

File inodes and the inode structure. File links – hard and soft links. Filters. Head and tail commands. Cut and paste commands. The sort command and its usage with different options. The umask and default file permissions. Two special files /dev/null and /dev/tty. Meaning of a process. Mechanism of process creation. Parent and child process. The ps command with its options. Executing a command at a specified point of time: at command. Executing a command periodically: cron command and the crontab file.. Signals. The nice and nohup commands. Background processes. The bg and fg command. The kill command. The find command with illustrative example.

Self Study: Comparison of different operating systems

TEXT BOOKS:

- Sumitabha Das., Unix Concepts and Applications., 4th Edition., Tata McGraw Hill
- .Behrouz A. Forouzan, Richard F. Gilberg : UNIX and Shell Programming- Cengage Learning – India Edition 2009
- M.G. Venkatesh Murthy: UNIX & Shell Programming, Pearson Education.
- Richard Blum , Christine Bresnahan : Linux Command Line and Shell Scripting Bible, 2nd Edition Wiley 2014

BLUE PRINT FOR QUESTION PAPER

UNITS	2 MARKS (10 Questions)	6 MARKS (7 Questions)	10 MARKS (3 Questions)	TOTAL
UNIT 1	2	1	-	10
UNIT 2	2	1	1	20
UNIT 3	2	2	1	26
UNIT 4	2	1	-	10
UNIT 5	2	2	1	26
TOTAL MARKS	20	42	30	92

TITLE : DATA COMMUNICATION AND COMPUTER NETWORKS

CODE : CA4318

Hrs / Week : 4 Hrs

CREDITS 4

Course Objectives

Become familiar with the basics of computer networks
Become familiar with network architectures
Become familiar with fundamental protocols
Become familiar with basic network computing techniques

Learning Outcomes

Upon completion of this module, students will be able to:
Have a good understanding of the OSI Reference Model and in particular have a good knowledge of Layers 1-3.
Analyze the requirements for a given organizational structure and select the most Appropriate networking architecture and technologies

UNIT I: **(12)**

DATA COMMUNICATION

Components of Data Communication – Direction of Data flow – Analog and digital signals ,its properties, Networks – Components and Categories – Types of Connections – Topologies –Protocols and Standards – ISO / OSI model, Example Networks such as ATM, Frame Relay, ISDN Physical layer: Transmission modes, Multiplexing, Transmission Media, Switching, Circuit Switched Networks, Datagram Networks, Virtual Circuit Networks.

UNIT II **(12)**

DATA TRANSMISSION SYSTEMS

properties of media and digital transmission Systems – Twisted Pair , Coaxial Cable, Optical Fibre, Radio Transmission Infrared Light Error detection and correction – Error detection , Two dimensional parity checks , Internet checksum , Polynomial code; standardized Polynomial codes , Error detecting capability of a polynomial code, Multiplexing – frequency – Division , Time Division , SONET; Wavelength Division Multiplexing Circuit switches; Telephone network , signaling Traffic and Overload control in Telephone networks – Concentration, Routing Control, Overload controls Cellular Telephone Networks, Satellite Cellular networks.

UNIT III: **(12)**

DATA LINK LAYER

Introduction, Framing, and Error – Detection and Correction – Parity – LRC – CRC Hamming code, Flow and Error Control, Noiseless Channels, Noisy Channels, HDLC, Point to Point Protocols. 111 Medium Access sub layer: ALOHA, CSMA/CD, LAN – Ethernet IEEE 802.3, IEEE 802.5 – IEEE 802.11, Random access, Controlled access, Channelization

UNIT IV : **(12)**

NETWORK LAYER

Logical Addressing, Internetworking, Tunneling, Address mapping, ICMP, IGMP, Forwarding, Uni-Cast Routing Protocols, Multicast Routing Protocols, Dynamic routing: RIP,OSPF,BGP,CIDR.

UNIT V: **(12)**

TRANSPORT LAYER AND APPLIATION LAYER

Process to Process Delivery, UDP and TCP protocols, IP,TCP,UDP headers Data Traffic,Congestion, Congestion Control, QoS, Integrated Services, Differentiated Services, QoS in Switched Networks. **:** Domain name space, DNS in internet, electronic mail, SMTP, MIME,FTP,WWW, HTTP, SNMP,NFS protocols.

Self Study: DNS

TEXT BOOKS:

- Data Communications and Networking, Behrouz A. Forouzan , Fifth Edition .
- Data and Computer Communication, William Stallings, Eigth Edition, Pearson Education,
- ComputerNetworks, Andrew S Tanenbaum, 5th Edition. Pearson Education, PHI.
- An Engineering Approach to Computer Networks, S. Keshav,2nd Edition, Pearson Education.
- Understanding communications and Networks, 3rd Edition, W.A. Shay, Cengage Learning.

BLUE PRINT FOR QUESTION PAPER

UNITS	2 MARKS (10 Questions)	6 MARKS (7 Questions)	10 MARKS (3 Questions)	TOTAL
UNIT 1	2	1	-	10
UNIT 2	3	2	-	18
UNIT 3	2	2	1	26
UNIT 4	1	1	1	18
UNIT 5	2	1	1	20
TOTAL MARKS	20	42	30	92

TITLE : WEB DEVELOPMENT

CODE : CAOE4418 (Open elective for other students)

Hrs / Week : 2 Hrs

CREDITS 2

Course Objectives: To teach developing static and dynamic web sites with hands on practical experience.

Learning Outcome:

On successful completion of the course the students will be able to do the following:

- To provide an in-depth training for web development skills.
- To understand and develop l web pages independently.
- To understand the methods of debugging and correcting anomalies.
- To provide a proper foundation for learning other tools of web development.

UNIT I **(10)**

INTERNET BASICS

Introduction to internet and its applications, E-mail, telnet, FTP, E-commerce, video conferencing, e-business. Internet service providers, domain name server, internet address, World Wide Web and its evolution, uniform resource locator (URL), browsers – internet explorer, netscape navigator etc. search engine, web saver – apache, proxy server, HTTP protocols

UNIT II **(10)**

WEB DESIGNING TECHNOLOGIES

Introduction to HTML5, ASP, JSP, Java scripts, VB scripts, HTML5/DHTML – file creation, HTML tags, titles footers, text formatting, forms, images, lists, tables, linking documents, front page editing, frame sets, order list, un-order list, special characters

UNIT III **(10)**

JAVA SCRIPT

Introduction to Java script, writing java script into HTML, Building of Java Script Syntax Data types of variables, arrays, operators, expressions, programming construct of conditional checking, loop ends functions, dialogue boxes

CSS:

Understanding the importance of CSS, Types: inline, internal and external with examples.

Self Study: Lists

TEXT BOOKS

- E. Balaguruswamy, Programming with JAVA, A Primer, 4th Edition., TMH .
- KenArnold & James Gosling, The Java Programming Language, Addison – Wesley,
- Patrick Naughton & Herbert Schildt, JAVA 2: The Complete Reference.

BLUE PRINT FOR QUESTION PAPER

UNITS	3 MARKS (5 QUESTIONS)	5 MARKS (4 QUESTIONS)	TOTAL
UNIT 1	1	1	8
UNIT 2	1	2	13
UNIT 3	1	1	8
UNIT 4	1	1	8
UNIT 5	1	1	8
TOTAL MARKS	15	30	45

PRACTICALS

TITLE : JAVA PROGRAMMING LAB

CODE : CA4P1

Hrs / Week : 2 Hrs

CREDITS 1

1. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result
2. To accept 5 subject marks through command line arguments , find the average and total of the mark. Display the result in various grades as follows.
3. Create one single dimensional array type of string and display the text in alphabetical order.
4. Generate a multi level inheritance program which used to demonstrate constructor overloading.
5. Generate a java program which shows the difference between static, final,, abstract access modifiers.
6. Create one object array to store minimum 50 students database.
7. Create one interface with all arithmetic operations and implement it to demonstrate Interface implementation.
8. Create one package to operate on all arithmetic operations and import those methods in normal java program.
9. To do the following operations on the given set of strings. a)concatenation. b) Comparison c) Character extraction. d)Length of string. use string buffer to generate the list of string operations.(any 7 functions)
10. Create a java program to explain multiple try and nested try block statements.
11. Create your own exception to handle the exception when the input value is above 10
12. Generate one single thread. a)using Thread class
    b) using Runnable Interface.
13. To sort list of elements in ascending and descending order and show the exception handling.
14. To implement constructor overloading by passing different number of parameter of different types.
15. To create student report using applet, read the input using text boxes and display the o/p using buttons.
16. To calculate bonus for different departments using method overriding.
17. To demonstrate Mouse Events
18. To Demonstrate Keyboard Events
19. To Implement Graphics

TITLE : UNIX LAB

CODE : CA4P2

Hrs / Week : 2 Hrs

CREDITS 1

1. Study of Unix Internal And external commands.
2. Study of File and Directory related commands.
3. Study of Vi editor.
4. Study of pattern searching.
5. Study of connecting commands.
6. Study of redirection commands.
7. Study of Regular Expressions.
8. Study of Shell Programming.
9. Study of process commands.
10. Study of Filter Commands.
11. Study of Background Processing.
12. Study of Filter Commands.

SYLLABUS FOR BACHELOR OF COMPUTER SCIENCE (BCA) COURSE

Re-accredited with 'A++' GRADE and

3.79/4 CGPA by NAAC Recognized by

UGC as College of Excellence

Implemented from 2018 -19 Batch

Course Structure and Syllabus for

Bachelor of Computer Applications (BCA)

(Six Semester Course)

Proposed for 2018 batch onwards

With

CHOICE BASED CREDIT SYSTEM (CBCS)

Department of Computer Science
Faculty of Science
St. Joseph's College (Autonomous)

FIFTH SEMESTER

Theory							
Sl.No.	**Code**	**Course Title**	**Hrs / Week**	**Credits**	**IA**	**SE**	**Total**
1	CA5118	Web Technologies	4	4	30	70	100
2	CA5218	System Software	4	4	30	70	100
3	CA5318	Computer Graphics and Multimedia	4	4	30	70	100
4	CA5415	Operations Research	4	4	30	70	100
Practical							
1	CA5P1	Web Technologies and Computer Graphics Lab	2	1	70	30	100
2	CA5P2	Mini Project Lab	2	1	70	30	100

SIXTH SEMESTER

Theory							
Sl.No.	**Code**	**Course Title**	**Hrs / Week**	**Credits**	**IA**	**SE**	**Total**
1	CA6118	Computer Organization and Architecture	4	4	30	70	100
2	CA6218	Object Oriented Analysis and Design using Unified Modeling Language	4	4	30	70	100
3	CA6318	Software Testing and Quality Assurance	4	4	30	70	100
4	CA6418	Mobile Applications	4	4	30	70	100

Practical							
1	**CA6P1**	**Mobile Applications Lab**	**2**	**1**	**70**	**30**	**100**
2	**CA6P2**	**Major Project Lab**	**2**	**1**	**70**	**30**	**100**

TITLE: WEB TECHNOLOGIES
PAPER CODE: CA 5118
NO OF HOURS: 60
CREDITS: 4

Objectives

1. To know and understand the concepts of internet programming.
2. To understand the concepts of HTML, CSS, BOOTSTRAP, PHP and MYSQL.
3. To develop own Web page and to host own web site on internet.

Learning Outcome:

After studying that subject students would have capability to make own web site and host their own web site on internet. Also students would have enough knowledge about what are the technologies used in internet.

UNIT: 1

12 hrs

Fundamentals
Internet, WWW, Web Browsers and Web Servers, URLs, MIME, HTTP, Security, Evolution of the Web, Internet Applications, Networks, TCP/IP, Search Engines.

HTML
Introduction to HTML, HTML Tags, Text formatting, Forms, Images, Lists, Tables, Linking documents, Frame sets, Lists

Cascading Style Sheets (CSS)
Types of styles, CSS selectors, Span & div tags, header styles, Text and font attributes, Backgrounds, Box model

UNIT: 2

12 hrs

Java Script

Introduction to Java script, writing java script into HTML, Data types , variables, arrays, operators, programming construct of conditional checking, loop ends functions, dialogue boxes, Objects, events, DOM, Differences between ES5 and ES6

UNIT: 3

12 hrs

Bootstrap: Introduction, Advantages of bootstrap framework, Applying bootstrap framework to applications, Bootstrap Grid: What is bootstrap grid? Bootstrap containers and its types, Displaying responsive images in bootstrap, Using readymade themes in bootstrap, Tables, form layout, button, text ,Bootstrap components.

UNIT 4

12 hrs

Introduction to PHP

Introduction to PHP, Language basics, Variable, constant, keywords, Data Types, Control Structures, Type casting, $_GET, $_POST,

Function and string in PHP

Defining and calling a function, Types of strings in PHP,

Arrays in PHP

Identifying elements of an array, storing data in arrays, Multidimensional arrays

UNIT 5

12 hrs

MYSQL

Introduction to MySQL- select statements, creating database/tables, inserting values, updating and deleting, PHP with MySQL, creating connection, selecting database, file handling in PHP – reading and writing from and to FILE, Using MySQL from PHP.

References:

1. Steven Holzner,"HTML Black Book", Dremtech press.
2. Web Technologies, Black Book, Dreamtech Press

3. Ivan Bayross, “Web Technologies Part II”, BPB Publications.
4. Programming PHP - RasmusLerdorf and Kevin Tatroe, O'Reilly publication, (3rd Edition).
5. HTML5 Pocket Reference (O'Reilly)
6. Bootstrap Site Blueprints Volume II
7. Advanced Detailed Approach to Master PHP Programming Language for Web Development
8. MySQL Stored Procedure Programming

BLUE PRINT OF THE QUESTION PAPER

UNITS	2 MARKS (10 Questions)	6 MARKS (7 Questions)	10 MARKS (3 Questions)	TOTAL MARKS
1	2	2	1	26
2	2	1	-	10
3	2	2	1	26
4	2	1	1	20
5	2	1	-	10
TOTAL	**10**	**7**	**3**	**92**

TITLE: SYSTEM SOFTWARE
Paper Code: CA5218
No of Hours: 60 Hrs.
Credits: 4

Objectives:

1. To understand the execution process of HLL programs.
2. To understand the working of scanners and parsers
3. To understand the basic design of various system software.
4. To implement various system software.

Learning Outcome:

Able to:

- Study the architecture of a hypothetical machine, its assembly language, macro language.
- Program in assembly language.
- Understand the structure and design of assemblers, linkers and loaders.

- Understand the concepts and theory behind the implementation of high level programming languages.

Unit -1
FUNDAMENTAL TO SYSTEM SOFTWARE, MACHINE STRUCTURE, MACHINE LANGUAGE AND ASSEMBLY LANGUAGE:
12**Hrs**
Background: Machine Structure, Evolution of the Components of a Programming System, Assembler, Loaders, Macros, Compliers, Formal Systems. Machine Structure, Machine Language and assembly language: General Machine Structure, Machine Language, Assembly Language

Unit-2
ASSEMBLERS:

12**Hrs**
Introduction to Assembler ,General Design Procedure, Design of assembler, Statement of Problem, Data structure, Format of data structure, algorithm, look for modularity, Table Processing: Sorting, Shell Sort, Bucket Sort, Radix Exchange Sort, address calculation sort, comparison of sorts, hash or random entry searching.

Unit-3
MACRO LANGUAGE AND THE MACRO PROCESSOR:

12**Hrs**
Introduction to Macros, Macro instruction, Features of macro facility, Macro instruction arguments, conditional macro Expansion, and macro calls within macros, macro instructions defining macros, Implementation, Statement of problem, implementation of a restricted facility, a two pass algorithm, a single pass algorithm, implementation of macro calls within macros. Implementation within an assembler.

Unit -4
LOADERS:

12**Hrs**
Introductions to loaders, Loader schemes, Compile & go, General loading Scheme, absolute loaders, Subroutine Languages, Relocating loaders, Direct linking loaders, other loading Schemes – Binders, linking loaders, Overlays, Dynamic binders. Design of absolute loader., Design of a Direct linking loader, Specification of problem, Specification of data structure, format of data bases ,algorithm.

Unit-5
COMPILERS:

12**Hrs**

Statement of problem, Regular expressions, Context Free Grammar, Problem1: Recognizing basic Elements, Problem2: Recognizing Syntactic cutis & interpreting meaning, Problem3: Storage Allocation, Problem4: Code Generation. Optimization (machine independent) optimization (machine dependent), General Model of complier, Simple Structure of Compiler, Brief introduction to 7 Phases of Compliers.

Self-Study: Sorting examples, program and examples of Macros, examples representing different types of compliers.

Text Book:

1. Chitra Ravi: System Programming, BCA 6th sem BU

Reference:

1. Jeffrey Ullman: Basics of Compiler Design.
2. John J. Donowon, System Programming, TATA McGraw-Hil.
3. Beck: System Software, 3/e Pearson Education
4. Dhamdhere: System Programming and Operating System TMH
6. Laudon & Laudon, Management Information Systems, 8/e. Pearson Education.

BLUE PRINT OF QUESTION PAPER:

Units	**2 marks**	**6 marks (7 questions)**	**10 marks (3 questions)**	**Total**
Unit1	2	1	½	13
Unit 2	2	1	½	13
Unit 3	2	2	1	24
Unit 4	2	2	½	19
Unit 5	2	2	½	19
Total	20	42	30	**92**

TITLE: Computer Graphics and Multimedia.
Paper code: CA5318
No of Hours: 60
Credits: 4

Objectives:
This course deals with two and three dimensional transformation, projection and graphical functions. It helps to have a better understanding of 2D and 3D technologies.

Learning Outcomes:
Able to:

- Describe the fundamental algorithms used in computer graphics.
- Work and interact through hands-on experiences to design, develop and modify electronically generated imagery using a wide range of sophisticated graphical tools and techniques.
- Summarize different hidden surface elimination Evaluate algorithms and shading techniques used in computer graphics and digital media production.
- Explain about the technology necessary for creating multimedia content for the web, video, DVD, 2D and 3D graphics, sound and programming.
- Understand the knowledge, techniques, skills and modern tools to become successful professionals in communication and Media industries.

Unit -1

Introduction to computer graphics, graphics systems and Devices:
10hrs

Overview of computer graphics, representing pictures, preparing, presenting & interacting with pictures for presentations; Visualization and image processing; RGB color model, direct coding, lookup table, storage tube graphics display, Raster scan display.

Self-Study: Applications, 3D viewing devices, Plotters, printers, digitizers, Light pens etc.; Active & Passive graphics devices; Computer graphics software.

Unit -2

Scan conversion Points, lines and Circle
12 Hrs

Line drawing algorithms: DDA algorithm and Bresenham's line algorithm. Circle generation algorithms, Ellipse generating algorithm, scan line polygon, fill algorithm, boundary fill algorithm, Flood fill algorithm.

Self-Study: Examples all algorithms.

Unit -3

12Hrs

2D transformation Basic transformations and viewing

translation , rotation, scaling ; Matrix representations & homogeneous coordinates, transformations between coordinate systems; reflection shear; Transformation of points, lines , parallel lines, intersecting lines. Viewing pipeline, Window to viewport Co-ordinate transformation, clipping operations, point clipping, line clipping, clipping circles, polygons & ellipse.

Self-Study: Coordinate Geometry, clipping examples.

Unit -4

12Hrs

3D transformation & viewing 3D transformations:

Translation, rotation, scaling & other transformations. Rotation about an arbitrary axis in space, reflection through an arbitrary plane; general parallel projection transformation; clipping, viewport clipping, 3D viewing:

Curves: Curve representation, surfaces, designs, tree representation.

Hidden surfaces Depth comparison: Z-buffer algorithm, Back face detection, scan-line algorithm; Hidden line elimination.

Self-Study: examples of 3d transformations.

Unit-5

Multimedia

14

Hrs

Introduction to Multimedia: Concepts, hypertext and hypermedia. Image, video and audio standards, Audio: digital audio, MIDI, processing sound, sampling, compression. Video: MPEG compression standards, compression through spatial and temporal redundancy, inter-frame and intra-frame compression. Animation: types, techniques, key frame animation, utility, morphing. Virtual Reality concepts.

Case study: uses of multimedia and Application.

Text book:

1. Introduction to Computer Graphics and Multimedia by Anirban Mukhopadhyay and Arup Chattopadhyay.
2. Andleigh, P. K and Kiran Thakrar, "Multimedia Systems and Design", PHI, 2003

Reference:

1. Judith Jeffcoate, "Multimedia in practice: Technology and Applications", PHI, 1998.
2. Foley, Vandam , Feiner and Huges, "Computer Graphics: Principles and Practice", 2nd Edition, Pearson Education, 2003.

BLUE PRINT OF QUESTION PAPER:

Units	**2 marks**	**6 marks (7 questions)**	**10 marks (3 questions)**	**Total**
Unit1	2	1	1/2	15
Unit 2	2	1	1/2	15
Unit 3	2	1	1	20
Unit 4	2	2	1/2	21
Unit 5	2	2	1/2	21
Total	20	42	30	**92**

TITLE: OPERATIONS RESEARCH
Paper code: CA5418
Total Teaching Hours: 60
Credits: 04

Objectives:

This course aims at teaching solution techniques for solving linear programming models, simple queuing model, two-person zero sum games and Network models.

Learning Outcome:

Able to:

- Solve Linear Programming Problems using Simplex Algorithm.
- Solve Transportation and Assignment Problems
- Understand different types of Queuing Models and solve related problems
- Obtain the solution of two person zero sum games using Linear Programming
- Formulate Maximal Flow Model using Linear Programing.
- Use PERT and CPM.

UNIT -1

12 hrs

LINEAR PROGRAMMING PROBLEMS:

Origin and development of operations research, Linear Programming Problem –formulation of Linear Programming problem, Graphical solution. Theory of simplex method. Use of artificial variables and their solution.

Self-Study: Simplex method examples, LPP history and applications.

UNIT - 2

12 hrs

TRANSPORTATION PROBLEM:

Mathematical formulation of transportation problem, Initial basic Feasible solution, North West corner rule, Matrix minima method, Vogel's approximation method, MODI method to find optimal solution.

Self-study: Vogel's' approximation method Examples, Transportation strategies.

UNIT - 3

12 hrs

ASSIGNMENT PROBLEM:

Mathematical formulation of an Assignment problem, Assignment algorithm, Hungarian Method to solve Assignment Problem.

UNIT - 4

12 hrs

NETWORK ANALYSIS:

Basic components of Network, Rules for drawing Network diagram Time calculation in Networks. Critical Path Method and PROJECT Evaluation and Review Techniques. Algorithm and flow chart for CPM and PERT.

Self –study: Algorithm approach and study on PERT and CPM.

UNIT - 5

12 hrs

THEORY OF GAMES:

Two–person Zero–sum Games, the Maxi-min and Mini-max principle, Saddle point and Value of the Game. Game without saddle points, mixed strategies, solution for 2X2 games, Graphical method Dominance property.

Self -study: gaming Strategies.

TEXT BOOKS:

1. Taha, "Operations Research", 7th edition, Pearson Education, 2007.

REFERENCES:

1. Billey E. Gillett, "Introduction to Operations Research", Himalaya Publishing House, Delhi, 1979.

2. Hamady A.Taha "Operations Research", Collin Mac Millan, 1982.

BLUE PRINT OF QUESTION PAPER:

UNITS	**2 MARKS (10 Questions)**	**6 MARKS (7 Questions)**	**10 MARKS (3 Questions)**	**TOTAL MARKS**
UNIT 1	2	2	1	26
UNIT 2	2	1	1	20
UNIT 3	2	2	1	26

UNIT 4	2	1	-	10
UNIT 5	2	1	-	10
TOTAL	**10**	**7**	**3**	**92**

TITLE: WEB TECHNOLOGIES AND COMPUTER GRAPHICS LAB
PAPER CODE: CA5P1
CREDIT: 1

WEB TECHNOLOGIES: A Website is to be developed using different concepts decided by the teacher.

GRAPHICS:

1. Program to Implement DDA Line Drawing Algorithm.

2. Program to Implement Bresenham's Line Drawing Algorithm.

3. Program to Implement DDA Circle Drawing Algorithm.

4. Program to Implement Bresenham's Circle Drawing Algorithm.

5. Program to draw a line in 4 quadrants according to user's Input.

6. Program to Implement Translation of a Rectangle, Triangle or a Circle in 4 quadrants according to user's choice.

7 Program to Implement Scaling of a Rectangle, Triangle or a Circle in 4 quadrants according to user's choice.

8. Program to Implement Rotation of a Rectangle, Triangle or a Circle in 4 quadrants according to user's choice.

9. Program to perform Composite Scaling.

10. Program to implement Composite Transformation.

11. Program to Implement Flood Fill Algorithm.

12. Program to Implement Boundary Fill Algorithm.

TITLE: MINOR PROJECT LAB
PAPER CODE: CA 5P2
CREDIT: 1

GUIDELINES

- The aim of the project work is to acquire practical knowledge on the implementation of the programming concepts.
- Each student should carry out individually which is involving in a group of maximum of four students and it may be a work using the software packages that they have learned or the implementation of concepts from the papers studied or implementation of any innovative idea focusing on application oriented concepts.
- The project work should be compulsorily done in the college only under the supervision of the department staff concerned.

The following are the suggested projects using the different areas of software concepts Python, VB, C#, Java. The project will be Windows based along with Database.

Sl. no	**Topic Name**
1	Online Tax Information System.
2	Online mobile phone shop
3	Employee-Management-System
4	Student Management system
5	Hotel Management System
6	Payroll Management System.
7	Library Management System.
8	Project on Airline Reservation System
9	Airline Ticket Booking System Project
10	Network Banking System
11	Healthcare Hospital system
12	Admission procedure
13	Time table generation
14	Examination result according to the classes

MINOR PROJECT EVALUATION FORMAT

Scheme of Valuation

Internal

1. Average for all Eleven labs , each lab carries 10 marks 20 marks
2. Verification of Report with Explanation 10 marks

Total Internal 30 Marks

External

1.	Demonstration and presentation	50 marks
2.	Documentation	20 marks

Total External 70 Marks
Total Marks 100 Marks

SIXTH SEMESTER BCA

TITLE: Computer Organization And Architecture
Paper Code: CA 6118
No of hours: 60Hrs
CREDITS: 4

Objective:

- To enable the students to learn the basic functions, principles and concepts of Computer Architecture.
- To learn the fundamental aspects of Computer Architecture and design.

- To focus on processor design, control unit design techniques.
- To study on I/O interfacing.
-

Learning Outcome:

Able to:

- Understand Computer Architecture.
- Understand I/O, Registers and memory.
- Understand processor design, control unit design.
- Understand I/O interfacing.

Unit 1

Computer Evolution, Integrated circuits and error detection codes:

12 Hrs

Functional units of a computer, Basic operational concepts, Bus structure, Addressing methods, Memory locations and addresses, Instructions and instruction sequencing, Instruction execution**,** Von Neumann Architecture, digital integrated circuits, IC flip-flops and registers, decoders and multiplexers, binary counters, shift registers, random - access memories (RAM) read-only memories (ROM), fixed-point representation, floating-point representation, other binary codes, error detection codes.

Unit 2

Central Processing Unit, Instructions and organization:

12 Hrs

Instruction Characteristics, Types of Operands, Types of Operations, Addressing Modes, Instruction Formats, timing and control, fetch cycle execution and instruction cycle, input output and interrupt, design of computer, Processor bus organization, arithmetic logic unit (ALU), stack organization, instruction formats, addressing modes, data transfer and manipulation, program control, microprocessor organization.

Unit 3

Input-output organization:

12 Hrs

Peripheral devices, Need of I/O Module, External Device , Input / Output Module, I/O Module Function, I/O Module Decisions, Input Output Techniques, Programmed I/O, I/O commands, I/O instructions, Interrupt Driven I/O, Basic concepts of an Interrupt , Response of CPU to an Interrupt, Design Issues, Priorities, Interrupt handling, Types of Interrupts, direct memory access (DMA), input-output processor (IOP), CPU –IOP communications.

Unit 4

Memory organization:

12 Hrs.

Characteristics of Memory Systems, Main Memory, Static and dynamic memories, Design of memory subsystem using Static Memory Chips, Design of memory subsystem using Dynamic Memory Chips, Memory interleaving, Cache Memory, Principles of cache memory, Structure of cache and main memory, Performance using cache memory, Elements of Cache Design, Mapping functions, Replacement algorithms, External Memory, Virtual memory, Memory Management in Operating Systems.

Unit 5

Multiprocessor Configuration:

12 Hrs.

Multiprocessing, Advantages and disadvantages of multiprocessing, Multiprogramming vs. Multiprocessing, Coupled Multiprocessor, Closely coupled multiprocessor, Loosely coupled multiprocessor, Comparison between closely coupled and loosely coupled Multiprocessor, Contention problems in multiprocessor systems, Memory contention, Communication Contention, Hot Spot contention, Techniques for reducing contention, Coprocessor, I/O Processor.

Self-study: Bus System, Examples of addressing modes,

Text book :

COMPUTER SYSTEM ARCHITECTURE – by M. MORRIS MANO - 3rd Ed

Reference:

1. “Computer Organization and Design: The Hardware/Software Interface” by David A. Patterson and John L. Hennessy
2. “Computer Organization” by Carl Hamachar, Zvonco Vranesic and Safwat Zaky
3. “Computer Architecture and Organization” by John P. Hayes
4. “Computer Organization and Architecture: Designing for Performance” by William Stallings
5. “Computer Systems Design and Architecture” by Vincent P. Heuring and Harry F. Jordan TITLE : Multimedia Technology.

BLUE PRINT OF QUESTION PAPER:

Units	2 marks	6 marks	10 marks	Total

	(10 questions)	(7 questions)	(3 questions)	
Unit1	2	1	1/2	15
Unit 2	2	1	1/2	15
Unit 3	2	1	1	20
Unit 4	2	2	1/2	21
Unit 5	2	2	1/2	21
Total	20	42	30	**92**

TITLE: OBJECT ORIENTED ANALYSIS AND DESIGN (OOAD) WITH UML
Paper Code: CA 6218
NO OF HOURS: 60 Hrs.
CREDITS: 4

Course Objective:

Object Oriented Analysis and Design Using UML Course Description: provides instruction and practical experience focusing on the effective use of object-oriented technologies and the judicious use of software modeling as applied to a software development process.

Learning outcomes:

Able to:

• Describe Object Oriented Analysis and Design concepts and apply them to solve problems.

• Prepare Object Oriented Analysis and Design documents for a given problem using Unified Modelling Language.

Unit -1

Complexity, The Object Model:

12 Hrs

Complexity: The inherent complexity of software, The Structure of complex systems, Bringing order to chaos, on designing complex systems, Categories of analysis and Design methods. The Object Model: The evolution of object model, Elements of object model, applying the object model, foundations of the object model.

Self-study: object, Notations of object.

Unit -2

Classes and Objects, Classification:

12 Hrs

Classes and Objects: The nature of an object, Relationship among objects, the nature of a class, Relationship among classes, the interplay of classes and objects, On building quality classes and objects, invoking a method. Classification: The importance of proper classification, Identifying classes and objects, Key abstraction and mechanisms, a problem of classification.

Self- study: class notation, quality classes and method.

Unit- 3

Notations with OOAD:

12 Hrs

Basic Behavioural Modeling, Basic elements, class diagram, object, state Transition diagram, Interactions, Use Case Diagrams, Activity, module and process diagrams.

Self-study: ATM using all Diagrams.

Unit -4

Process:

12 Hrs

Principles, Micro and macro development process, Pragmatics- Management and planning, staffing, Release management, Reuse, Quality Assurance Metrics, Documentation, Tools, The benefits and risks and Object-oriented development.

Self-study: Difference micro and macro and Risk strategies.

Unit -5

Architectural Modeling:

12 Hrs

Components, Deployment, Collaborations, Pattern and Frameworks, Component Diagram, Deployment Diagrams, Systems and Models. Case Study: A domain based analysis and design using Star UML (open source).

Self -study: using Components, Deployment, Collaborations, Pattern and Frameworks examples.

Text book:

1. Object oriented development system development by Ali Bahrami.

Reference:

1. Grady Booch, Object-Oriented Analysis And Design With Applications, Pearson Education, 3rd Edition, 2009.
2. Mahesh P Matha, Object Oriented Analysis and Design using UML, PHI, 3rd reprint, 2012
3. Grady Booch, James Rumbaugh and Ivar Jacobson, The Unified Modeling Languages User Guide, Addison Wesley, 4th Edition, Reprint 2000.
4. Mike O'Docherty, Object Oriented Analysis and Design Understanding system development with UML 2.0, John Wiley and Sons, 1st Edition.

BLUE PRINT OF QUESTION PAPER:

Units	**2 marks (10questions)**	**6 marks (7 questions)**	**10 marks (3 questions)**	**Total**
Unit1	2	1	1/2	15
Unit 2	2	1	1/2	15
Unit 3	2	2	1	26
Unit 4	2	1	1/2	15
Unit 5	2	2	1/2	21
Total	20	42	30	**92**

TITLE: SOFTWARE TESTING AND QUALITY ASSURANCE
PAPER CODE: CA 6318
NO OF HOURS: 60 HRS.
CREDITS: 4

OBJECTIVES:

- To enable students to learn different types of testing.
- To enable students to assess the quality of a software.

Learning Outcomes

Able to

- Utilize the different testing strategies to check the correctness of a software product.
- Demonstrate their capability to adopt quality standards.
- Assess the quality of software product.

UNIT I INTRODUCTION TO SOFTWARE TESTING
12 Hrs

Need for Software quality, role of testing, verification and validation, errors, faults, failure and defects, objectives of testing, test levels, white box and black box testing, test planning and design, test team organization, Basic concepts in testing theory-Theory of Goodenough and Gerhart-Theory of Weyuker and Ostrand-Theory of Gourlay, adequacy of testing, drawbacks of testing.

UNIT II TESTING STRATEGIES PART 1

18 Hrs

UNIT TESTING-Concept of Unit Testing-Static Unit Testing-Defect Prevention-Dynamic Unit Testing-Mutation Testing-Debugging -Junit Framework for unit testing, CONTROL FLOW TESTING-Outline of Control Flow Testing-Control Flow Graph -Paths in a Control Flow Graph-Path Selection Criteria-Generating. DATA FLOW TESTING- Data Flow Anomaly-Overview of Dynamic Data Flow Testing- Data Flow Graph- Data Flow Terms -Data Flow Testing Criteria -Comparison of Data Flow Test Selection Criteria -Feasible Paths and Test Selection Criteria -Comparison of Testing Techniques.

UNIT III TESTING STRATEGIES PART 2
18 Hrs

DOMAIN TESTING- Domain Error-Testing for Domain Errors-Sources of Domains-Types of Domain Errors. SYSTEM INTEGRATION TESTING-Concept of Integration Testing- Different Types of Interfaces and Interface Errors -Granularity of System Integration Testing -System Integration Techniques- Software and Hardware Integration. FUNCTIONAL TESTING-Different Types of Variables -Test Vector -Testing a Function in Context. ACCEPTANCE TESTING- Types of Acceptance Testing - Acceptance Criteria - Selection of Acceptance Criteria - Acceptance Test Plan

UNIT IV SOFTWARE QUALITY AND MATURITY MODELS
8 Hrs

Five Views of Software Quality- McCall's Quality Factors and Criteria-Quality Factors -Quality Criteria -Relationship between Quality Factors and Criteria -Quality Metrics, Basic Idea in Software Process, Capability Maturity Model- CMM Architecture - Five Levels of Maturity and Key Process Areas - Common Features of Key Practices - Application of CMM - Capability Maturity Model Integration(CMMI).

UNIT V STANDARDS, CERTIFICATIONS & ASSESSMENTS
4 Hrs

ISO 9126 Quality Characteristics, ISO 9000:2000 Software Quality Standard, ISO 9000:2000 Fundamentals, ISO 9001:2000 Requirements

Self-study: Different standards of quality checks done on a software product.

TEXT BOOK:

- Software testing and quality assurance by Kshirasagar Naik and Priyadarshi Tripathy, A JOHN WILEY & SONS, INC., PUBLICATION.

REFERENCES:

1. Software Testing: A Craftsman's Approach, Fourth Edition, by Paul C. Jorgensen.
2. Introduction to Software Testing(Edition 2) by Paul Ammann and Jeff Offutt.
3. Testing and Quality Assurance for Component-based Software,Mordechai Ben-Menachem , International Thompson Computer Press, 1997.
4. Quality assurance and software testing fundamentals-Liliana Lancu-Independent publication.

BLUE PRINT OF QUESTION PAPER:

Units	2marks (10 questions)	6marks (7 questions)	10marks (3 questions)	Total
1	2	1		10
2	3	2	1	28
3	3	2	1	28
4	1	1	1	18
5	1	1		8

Title: **MOBILE APPLICATIONS**

Code: **CA6418**

Total Hours: **60 Hrs**

Hours /week: **4**

Credits: **4**

Objectives

- To improve the speed by writing apps for Android devices
- Implementing layout management and multi-layout definition techniques to create adaptable user interfaces for mobile applications that share a common data model.
- Manage user data and multimedia on a mobile device via the Android framework libraries.
- Publish Apps to the Google Play Store.

Learning outcomes

- The student will be able to write simple GUI applications, use built-in widgets and components, work with the database to store data locally, and much more.
- Produce apps for mobile devices (iPhone/iPad/iPod Touch) and Gain a basic understanding of Android Architecture .
- The student will understand object-oriented programming - Develop a working knowledge of mobile app development tool.
- Understand mobile design principles and Identify need and opportunity in app markets.

UNIT 1: **ANDROID OVERVIEW AND ARCHITECTURE**

12 Hrs

Android History, Android versions, various Android devices on the market , applications , Android Development Environment - System Requirements, Android SDK, Installing Java, and ADT bundle - Eclipse Integrated Development Environment (IDE), Creating Android Virtual Devices (AVDs)

Android Architecture Overview and Creating an Example Android Application: The Android Software Stack, The Linux Kernel, Android Runtime - Dalvik Virtual Machine, Android Runtime – Core Libraries, Dalvik

VM Specific Libraries, Java Interoperability Libraries, Android Libraries, Application Framework.

UNIT 2: **ANDROID SOFTWARE DEVELOPMENT PLATFORM**

12 Hrs

Creating a New Android Project Defining the Project Name and SDK Settings, Project Configuration Settings, Configuring the Launcher Icon, Creating an Activity, Running the Application in the AVD, Activity and Activity Life cycle, Reviewing the Layout and Resource Files, Understanding Java SE and the Dalvik Virtual Machine , The Directory Structure of an Android Project , Common Default Resources Folders , The Values Folder , Leveraging Android XML, Screen Sizes , Launching Your Application: The AndroidManifest.xml File , Creating Your First Android Application

UNIT 3: ANDROID FRAMEWORK OVERVIEW AND VIEWS

12 Hrs

Android Application Components, Android Activities: Defining the UI, Android Services: Processing in the Background, Broadcast Receivers: Announcements and Notifications Content Providers: Data Management, Android Intent Objects: Messaging for Components Android Manifest XML: Declaring Your Components.

Understanding Android Views, View Groups and Layouts. Android Layout Managers, The View Hierarchy, Designing an Android User Interface using the Graphical Layout Tool.

UNIT 4: GRAPHICAL USER INTERFACE SCREEN, PICTURES AND MULTIMEDIA **12 Hrs**

Displaying Text with TextView, Retrieving Data from Users, Using Buttons, Check Boxes and Radio Groups, Getting Dates and Times from Users, Using Indicators to Display Data to Users, Adjusting Progress with SeekBar,

Working with Menus using views. Displaying PicturesGallery, ImageSwitcher, GridView, and ImageView views to display images, Creating Animation.

Multimedia: Audio, Video, Camera, Playing Audio and Video, Recording Audio and Video, Using the Camera to Take and Process Pictures.

UNIT 5: FILES, DATABASES, INTENT, ANDROID THREADS AND THREAD HANDLERS 12 Hrs

Saving and Loading Files, SQLite Databases, Android Database Design, Exposing Access to a Data Source through a Content Provider, Content Provider Registration, Native Content Providers

Intent Overview, Implicit Intents, Creating the Implicit Intent Example Project, Explicit Intents, Creating the Explicit Intent Example Application.

An Overview of Threads. The Application Main Thread, Thread Handlers, basic Threading Example, Creating a New Thread, Implementing a Thread Handler, Passing a Message to the Handler

SELF STUDY: Each unit's example apps have to be worked and executed in the class.

References

1. Professional Android 4, Recto Meier and Ian Lake., 4th Edition. (TEXT BOOK)

2. Android, A programmer's Guide, Jerome (J.F) DiMarzio.

2. Android Programming for Beginners, John Horton, 2015

3. Android Programming, The Big Nerd Ranch Guide, Second edition.

4. Android Development, Mark L Murphy, 4th Edition.

BLUE PRINT OF THE QUESTION PAPER

UNITS	2 MARKS (10 Questions)	6 MARKS (7 Questions)	10 MARKS (3 Questions)	TOTAL MARKS
UNIT 1	2	1	-	10

UNIT 2	2	2	1	26
UNIT 3	2	2	1	26
UNIT 4	2	1	1	20
UNIT 5	2	1	-	10
TOTAL	**10**	**7**	**3**	**92**

TITLE: MOBILE APPLICATION LAB
PAPER CODE: CA6P1
CREDIT: 1

List of programs

1. Implementing GUI components, Fonts and Colours.
2. Creating Simple login screen.
3. To illustrate Activity life cycle.
4. Creating user defined style and inheritance of styles
5. Working with Layout Managers and Event Listeners.
6. To create Native Calculator application in Android.
7. To illustrate the graphical primitive for Android App.
8. Creating Implicit intents
9. To work on Audio and video controls.
10. To illustrate the Running of SQLite for Android Application.
11. To work on external Data storage
12. To run Multithreading app.
13. To illustrate alert app in Android.
14. Implementing alarm clock app.
15. To work on GPS locator.

TITLE: MAJOR PROJECT LAB
PAPER CODE: CA6P2
CREDIT: 1

Project Guidelines

1) Understanding the significance of Project
Most of the students are under an impression that if good layout is proposed then it will help them in scoring good marks but the quality of project is analysed by proper write-ups. It provides an opportunity for learners to

demonstrate originality and to plan and organize the project work and put the practical approach of all the topics studied in the entire curriculum.

2) Meaning of Project

A project is a study of factual information for comprehending and applying the various concepts of the course into practice. Its main purpose is not to generalize but to study the situation with a practical orientation.

3) Steps in Project Formulation

a) System Study
b) System Analysis
c) System Designing
d) System Development
e) Implementation and Testing

4) Project Proposal

A proposal as per the format given should be prepared once the topic is selected. It should not be more than 3-4 pages and need not be sent separately. The format for the same is:

a) Title of Project
b) Objectives
c) Need for topic and Modules
d) Tools and Technology to be used in Project.
e) Methodology and Procedure of Work.

5) No Objection Certificate

If the project is carried out in a company or organization, then a certificate for no objection of same needs to be presented. It should mention that the organization has no objection in publishing the findings of the project study. The certificate should contain the name of authority with signature and company stamp and should be given on company's letterhead and duly signed by authorized signatory.

6**) Chapters**

Chapter 1: Study of Existing System and System Requirements

- What is the existing system?
- Define its scope.
- How the existing system works?
- What are the issues/problems with the existing system?
- How are you going to improve on the issues/problems with the existing system?

Chapter 2: Analysis

• Data flow Diagram
• Functional Decomposition

Chapter 3: Design
• ER diagram
• Data Dictionary
• Table Design
• Input forms design
• Report Layouts

Chapter 4: Coding
• Data Validations
• Sample Code

Chapter 5: Testing
• Different Testing implementations.

Chapter 6:
- Bibliography

7) Team size: Maximum of four students.
8) Technical Specifications of Project Report

Length: The length of the report should be between 80- 100 pages including the cover page, summary, table of contents, list of figures, list of tables, and acknowledgement.

Script and Page Format: The report should be typed using a Word Processor on standard A4 (210 mm x 297 mm) paper size. A conventional font, size 12-point and line spacing of 1.5 mm should be used.

Margins: Left-hand margins should have a width of not less than 38 mm to facilitate binding. The right hand, the top, and the bottom should be 25 mm. Each page must be typed in one side, leaving a wide margin.

Paper and Print Quality Paper and print quality: Paper and Print Quality Paper and print quality are important for successful legibility. The report can be printed on a standard quality paper, (e.g., photocopy paper)

Pagination: Positioning of page numbers should be on top right hand side. Pages starting from the summary until the last list of tables should be numbered using Latin numbers (I, II, III, IV…). Pages starting from the Introduction until the appendices should be numbered using numbers (1,2,3,…). Pages with figures and tables or illustrations must be also numbered.

Binding: The report should be hard bound.

Number of copies: Two copies should be submitted – self copy, College copy (Group).

Project Report Format Includes

1. Cover page
2. Certificate
3. Acknowledgment
4. Content
5. Abstract
6. Introduction
7. Study of Existing System
8. Study of Proposed System.
9. System Requirements
10. Analysis
11. Design
12. Coding
13. Testing strategies
14. Future Enhancement
15. Conclusion
16. Bibliography

MAJOR PROJECT EVALUATION FORMAT

Scheme of Valuation

Internal

1.	Average for all Eleven labs ,each lab carries 10 marks marks	20
2.	Verification of Documentation with Explanation marks	10
	Total Internal	**30 Marks**

External

1.	Demonstration presentation	and	50 mark
2.	Documentation		20 mark

Total External Marks **70**

Total Marks **100**
Marks