

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 Science
(Computer Science)
(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 Science is a three year course which consists of six semesters. It is a three major undergraduate programme with Computer Science as one of the subjects. The students are lead to realize the scope of computers which will arouse their curiosity to further research and continue their higher education in the field of computers. The newer concepts and the ideas are introduced to the students. They are also engaged in real time and advanced projects.

PROGRAM OBJECTIVE

The course is framed to the best learning of the technology in the present computing environment. It facilitates the students with the proper ambience and atmosphere for the sound learning of the concepts and ideas. It forms students to compete in the career and higher education. This course helps students to face challenges of the future.

SEMESTER WISE COURSE STRUCTURE

Semester	Code	Course title	Hrs/ week	Credits	IA	SE	Total
Theory							
I	CS118	Digital Computer Fundamentals and Programming in C	4	4	30	70	100
Practical							
I	CS1P1	C Programming lab	2	1	15	35	50
Theory							
II	CS218	Data structures and Operating systems	4	4	30	70	100
Practical							
II	CS2P1	Data structures lab	2	1	15	35	50
Theory							
III	CS318	Object Oriented Programming using Python and Database Management Systems	4	4	30	70	100
Practical							
III	CS3P1	Object Oriented Programming using Python and Database Management Systems	2	1	15	35	50
Theory							
IV	CS4118	.NET Technologies	2	2	15	35	50
Soft Core Paper (For Other Students)							
IV	CSOE4218	Web Development	2	2	15	35	50
Practical							
IV	CS4P1	.NET lab	2	1	15	35	50
Theory							
V	CS5118	JAVA Programming	3	3	30	70	100
	CS5218	Software Engineering	3	3	30	70	100
Practical							
V	CS5P1	JAVA Programming lab	2	1	15	35	50

	CS5P2	Web designing/ Mini Project lab	2	1	15	35	50
Theory							
VI	CS6118	Computer networks	3	3	30	70	100
	CS6218	Computer graphics and Multimedia	3	3	30	70	100
Practical							
VI	CS6P1	Major project lab	2	1	15	35	50
	CS6P2	Computer graphics lab	2	1	15	35	50

EXAMINATION AND ASSESMENTS

1. IA Weight age 30 %
2. End Semester Examination Weight age 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 | 10 marks |
| 2. | Execution of one program | 10 marks |
| 3. | Formatting the program and output | 05 marks |
| 3. | Record verification | 05 marks |
| 4. | Viva voce related to practical topics only | 05 marks |

Total	35 marks
-------	----------

PROJECT EVALUATION FORMAT

Scheme of valuation:

- | | | |
|----|--------------------------------|----------|
| 1. | Demonstration and presentation | 20 marks |
| 2. | Documentation | 15 marks |

Total	35 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

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.

15 marks

FIRST SEMESTER

Title:	DIGITAL COMPUTER FUNDAMENTALS AND PROGRAMMING IN C
Code:	CS118
Total Hours:	60 Hrs
Hours/Week:	4
Credits:	4

Objectives

Digital Computer Fundamentals

- To introduces the introductory concepts on computer hardware, software.
- To learn additional concepts which help us to understand upcoming technologies, number systems and its applications, Boolean algebra and different types of gate, networks.

Programming in C

- To give a practical exposure to problem solving in C programming.
- To study the algorithms and flowcharts
- To learn the programs and to solve problems through logical thinking using C.

Learning Outcome

- To understand the components and the working principle of a computer.
- To understand concepts of Boolean algebra, number systems and gates.
- To know the efficiency of combinational and sequential logic circuits.
- To understand the logic of solving a problem.
- To learn good Coding standards for writing programs.

DIGITAL COMPUTER FUNDAMENTALS

30 Hrs

UNIT 1

(10)

INTRODUCTION TO COMPUTERS

Functional block diagram of a digital computer, Generation of computers, Classification of Computers- Analog, Digital, Hybrid, Micro, Mini, Main frame computers etc.

REPRESENTATION OF DATA

Number Systems, and Inter-conversions among them, Binary arithmetic (Addition, Subtraction, multiplication, division) Binary number system complements- 1's and 2's

+
Complements subtractions, ASCII, Excess-3 code and Gray code, EBCDIC code and BCD code.

UNIT 2 **(8)**

BOOLEAN ALGEBRA AND LOGIC CIRCUITS

Boolean Algebra Laws and theorems, Gates- AND, OR, NOT, NAND, NOR, EXOR truth tables, Boolean expressions and their simplifications, SOP & POS- Karnaugh map simplification methods.

UNIT 3 **(12)**

COMBINATIONAL AND SEQUENTIAL CIRCUITS

Multiplexers, Demultiplexers, Decoders, Encoders, Half Adder, Full Adder, Parallel Adder-subtractor, Flip flops- RS, JK, D, T, Master Slave , Counters(Binary, modulus counters), Shift registers (Working principles only).

MEMORY

Memory hierarchy, ROM, RAM, Cache Memory and Organization of Cache Memory, Virtual memory, Addressing modes, Instruction format and instruction set.

PROGRAMMING IN C **30 HRS**

UNIT 4 **(10)**

INTRODUCTION TO PROGRAMMING

Problem Solving Using Computers- Language Classification, Problem Analysis, Algorithm and Flowchart design. Algorithms - Steps in developing algorithms, advantages and disadvantages. Flowcharts- Symbols used in developing flowcharts, advantages and disadvantages. Coding, testing, debugging, Documentation and maintenance. Program development and modular design.

INTRODUCTION TO C PROGRAMMING

History, C Conventions, Character Set, Identifiers, Keywords, Simple Data types, Modifiers, Variables, Constants, Operators, Operator precedence, Structure of a C program. Input and Output-Input and Output operation: Single character input and output, formatted input and output, buffered input and type casting.

UNIT 5

(10)

CONTROL STRUCTURES

Introduction, Conditional statement, if statement, if-else statement, nested if statement, else-if statement and switch statement. Goto statement. Looping statement, while statement, do-while statement, for statement, break and continue, nested for statement.

ARRAY

Introduction (One and two dimensional), Declaration of arrays, Initialization of arrays, processing with arrays. String manipulation, declaration of string, string operations.

FUNCTIONS

Introduction, advantages of subprograms, Function definition, function call, Actual and formal arguments, local and global variables, function prototypes, types of functions, recursive functions, arrays and functions.

UNIT 6

(10)

STORAGE CLASSES, STRUCTURES AND UNIONS

Introduction, types of storage classes, Introduction to structures, Advantages of structures, accessing elements of a structure, nested structures, array of structures, functions and structures, Unions, bit-fields, enumerated data types.

POINTERS

Introduction, pointer variable, pointer operator, pointer arithmetic, pointers and arrays, pointers and strings, array pointers, dynamic allocation.

TEXT BOOKS

- Fundamentals of Computers by Rajaraman, PHI(6th Edition)
- Digital Computer Fundamentals by Thomas C Bartee, McGraw Hill, VI Edition.
- ANSI C by Balaguruswamy. (6th Edition)

REFERENCES

- Computer Organization by Carl Hamacher V. Zaki, McGraw Hill.
- Digital computer Fundamentals by Malvino& Leach.
- Digital Computer Fundamentals by Malvino.
- Computers today by Sanders, Mcgraw Hill, (3rd Edition)
- C programming by Stephen C Kochan. , (3rd Edition)
- The C programming Language, Brian W Kernigham, Dennis M Ritchie, Prentice Hall Software Series.
- Spirit of C by Cooper.

BLUE PRINT OF THE QUESTION PAPER

UNITS	2 MARKS (10 Questions)	6 MARKS (7 Questions)	10 MARKS (3 Questions)	TOTAL MARKS
Digital Computer Fundamentals	5	3	1.5	43
Programming in C	5	4	1.5	49
TOTAL	10	7	3	92

PRACTICALS

Title: C PROGRAMMING LAB
Code: CS1P1
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

- Ability to 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

SECOND SEMESTER

Title:	DATA STRUCTURES AND OPERATING SYSTEMS
Code:	CS218
Total Hours :	60 Hrs
Hours/Week:	4
Credits:	4

Objectives

- To be able to practically implement the data structures like stack, queue and array
- To understand and implement different searching and sorting techniques.
- To introduce the concept behind the 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:

- Able to understand the need for Data Structures and integrating into Applications.
- Able to walk through insert and delete for different data techniques.
- Able to understand the basic working process of an operating system.
- Able to understand the importance of process management and scheduling.
- Able to understand the issues in process synchronization and memory management.

DATA STRUCTURES

30 HRS

UNIT 1

(10)

INTRODUCTION TO DATA STRUCTURES

Definition, Classification of data structures. Operations on data structures. Introduction to Time and Space complexity.

PRIMITIVE DATA STRUCTURES

Integer, Character, float, Strings, memory representation and primitive operations. String manipulations using pointers.

ARRAYS

Storage representation of 1D and 2D arrays. Insertion and deletion on 1D arrays, advantages and dis-advantages of arrays.

UNIT 2

(10)

STACKS AND QUEUES

Concepts, operations, sequential and linked implementation, Applications of stacks - recursion, tower of Hanoi, Infix to postfix, Evaluation of postfix expressions. Concepts of queues, operations, sequential and linked implementation, circular queues, Priority queues and Dequeues (Introductory Concept)- Application of queues.

UNIT 3

(10)

LINKED LISTS

Dynamic memory allocation, singly linked lists, operations on linked lists - Insertion and Deletion of a node. Introduction to circular linked list and doubly linked list.

TREES

Definitions and concepts-Binary trees, sequential and linked representation of Binary trees, Insertion and Deletion of binary trees, Binary tree traversals.

SEARCHING AND SORTING

Linear search and Binary search, Selection sort, Insertion sort and merge sort.

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

OPERATING SYSTEM

30 HRS

UNIT 4

(12)

INTRODUCTION TO OPERATING SYSTEMS

Introduction, Simple Batch systems, Multi-programmed Batched Systems, Time sharing systems, Personal Computer Systems, Distributed Systems and Real time Systems. Operating System structures, Command Interpreter System, Operating System Services, System calls and system programs.

PROCESS CONCEPT

Program and process, States of a process, Process control block, Process Scheduling, CPU Scheduling- Basic concepts, Scheduling criteria, Scheduling algorithms- FIFO, RR, SJF, Priority scheduling, Multi- level and Multi-level feedback.

UNIT 5

(8)

MEMORY MANAGEMENT

Basic Concepts, logical and Physical Address space, Swapping, Contiguous Allocation, paging, Segmentation, Virtual memory-Demand paging, page replacement and page replacement algorithm, allocation of frames, Thrashing and Demand Segmentation.

UNIT 6

(10)

FILE SYSTEM

File concept, access methods, directory structure, protection, File system structure, Allocation methods and free space Management.

I-O SYSTEMS

Overview of I/O systems, I/O interface, secondary storage structure -Disk structure, Disk Scheduling

TEXT BOOKS

- Data Structure by Schaum Series. (Revised 1ST Edition)
- Fundamentals of Data Structure by Horowitz Sahni(2nd Edition).
- Operating System by Milan Milenkovic, McGraw Hill. (5th Edition)

REFERENCES

- Data Structure by Dale and Lilly.
- Data Structures and Algorithm Analysis in C, Second Edition, Mark Allen Weiss, Pearson.
- 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
Data Structures	5	4	1.5	49
Operating Systems	5	3	1.5	43
TOTAL	10	7	3	92

PRACTICALS

Title: DATA STRUCTURES LAB
Code: CS2P1
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' efficiency and data structures in terms of time and memory complexity of basic operations.

Learning Outcome

- Student would be able to know the efficiency of 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. Merge Sort
5. Linear Search
6. Binary Search
7. Length of a string using pointer
8. Concatenate two strings using pointers
9. Copy a string using pointers
10. Array implementation of a stack.
11. Array implementation of a queue
12. Array implementation of circular queue.
13. Creating a linked list.
14. Adding nodes at various positions in a linked list.
15. Deleting nodes from various positions from a linked list.
16. Creating a binary search tree and performing the various traversals on a binary search tree.

Course Structure and Syllabus for

**Bachelor of Science
(Computer Science)
(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 Science is a three year six semester three major undergraduate programme with Computer Science as one of the subject. 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.

SEMESTER WISE COURSE STRUCTURE

Theory							
III	CS318	Object Oriented programming using Python and Database Management System	4	4	30	70	100
Practical							
III	CS3P1	Python and Database	2	1	15	35	50
Theory							
IV	CS4115	.NET	2	2	15	35	50
Soft Core Paper (For Other Students)							
IV	CSOE4218	Basic Programming skills	2	2	15	35	50
Practical							
IV	CS4P1	.NET lab	2	1	15	35	50

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	10 marks
2.	Execution of one program	10 marks
3.	Formatting the program and output	05 marks
3.	Record verification	05 marks
4.	Viva voce related to practical topics only	05 marks
	Total	35 marks

PROJECT EVALUATION FORMAT

Scheme of valuation:

1.	Demonstration and presentation	20 marks
2.	Documentation	15 marks
	Total	35 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 AND DATABASE MANAGEMENT SYSTEM

CODE : CS318

Hrs / Week : 4 Hrs

CREDITS 4

Course Objective:

- Comparison between procedural languages and object oriented languages.
- The concepts of operator overloading and function overloading, files, exceptions are discussed to expose the students to the advantages of object oriented programming.
- 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.
- Basic concepts of classes and objects make it easy to represent real world entities.

Learning Outcome:

- An understanding of the principles behind the object oriented development process.
- Competence in the use of object oriented programming language in the development of small to medium sized application programs.
- Understanding the core terms, concepts, and tools of relational database management systems. Understanding database design and logic development for database programming.

OBJECT ORIENTED PROGRAMMING USING PYTHON

30 HRS

UNIT I:

(10)

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:

(10)

INTRODUCTION TO OBJECT ORIENTED PROGRAMMING IN PYTHON

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

UNIT III:

(10)

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 .

DATA BASE MANAGEMENT SYSTEM

30 Hrs

UNIT IV

(12)

INTRODUCTION

Basic Concepts: Data, database, DBMS, Disadvantages of File oriented systems, Advantages of DBMS, database users, Database Languages, Characteristics of Database, Role of DBA, Data Abstraction (Views) – Logical, Conceptual & Physical, Data independence – physical and logical independence.

DATA MODELS

Introduction to Data Models: E-R model, Relational model, network model and hierarchical model.

UNIT V

(12)

RDBMS

Relational database concepts, attribute, tuple, types of attributes, single, multi-valued, stored, derived etc., keys, primary, index, candidate, alternate, foreign, Relationships, Relational algebra operations, union, intersection, difference, cartesian product, selection, projection, join, division, relational calculus, Normalization and its properties (1st, 2nd and 3rd and BCNF).

UNIT VI

(6)

DDL AND DML

DDL commands - create table/views/index, drop, alter, DML commands – select, insert, delete, update, etc., DCL commands – grant, revoke, commit, TCL commands, SQL – query, sub-query, nested query, Joins – natural, inner, outer join.

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

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
- Elmasri & Navathe, Fundamentals of Database Systems ,Pearson Education.
- Karate, Introduction to Database Management System, Pearson Education.
- Abrahamsi. Silberschatag, Henry. F. Korth, S. Sudarshan, Database System Concepts, McGraw hill.
- Data base System Concepts, Silberschatz, Korth, McGraw hill, Sixth Edition.
- An Introduction to Database systems, C.J. Date, A.Kannan, S.Swami Nadhan, Pearson, Eight
- Sundarraman, Oracle 9i programming A Primer,1/e Pearson Education.

BLUE PRINT FOR QUESTION PAPER

UNITS	2 MARKS (10 Questions)	6 MARKS (7 Questions)	10 MARKS (3 Questions)	TOTAL
UNIT 1	1	1	1/2	13
UNIT 2	2	1	-	10
UNIT 3	2	1	1	20
UNIT 4	1	1	-	08
UNIT 5	2	2	1/2	21
UNIT 6	2	1	1	20
TOTAL MARKS	20	42	30	92

PRACTICALS

TITLE : PYTHON AND SQL LAB

CODE : CS3P1

Hrs / Week : 2 Hrs

CREDITS 1

PRACTICALS

TITLE : PYTHON PROGRAMMING LAB

1. Demonstrate Running Python programs in Interactive Interpreter
2. Demonstrate Running Python programs in IDLE
3. Demonstrate Running Python programs in Command Prompt
4. To compute Distance between two points taking inputs from the user
5. Write a program to count the numbers of characters in the string and store them in a dictionary data structure
6. Write a program to use split and join methods in the string and trace a birthday with a dictionary data structure
7. Write a program to print each line of a file in reverse order
8. Write a program to compute the number of characters, words and lines in a file.
9. Write a program that defines a matrix and prints
10. Write a program to perform addition of two square matrices

TITLE : DATABASE LAB

1. To create a DDL to perform create, Alter, Modify the Table
2. To create a DDL to Drop a column
3. To create a DML to Insert a Single record into the table
4. To create a DML to Update the Table
5. To insert rows into the table using select clauses
6. To Select and Delete from the Table
7. To List the Records from the Table
8. To Implement Date and Time Functions
9. To Implement JOIN Queries
10. To Implement SET Operations

FOURTH SEMESTER

TITLE : .NET TECHNOLOGIES

CODE : CS4118

Hrs / Week : 2 Hrs

CREDITS 2

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 Visual Basic environment
- Define and implement form objects including data arrays, control arrays, text boxes, message boxes, dialog boxes, labels, controls, menus, frames, picture boxes, pull-down menus, and combo boxes. Use control array in your applications, Use the debugger in the Visual Basic environment to set breakpoints, program step, procedure step, and display any program variable
- Identify and perform the steps necessary to convert Visual Basic 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 Visual Basic.NET programs that meet commercial programming standards.
- Program design and coding is satisfactory.
- Work is 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:

(10)

INTRODUCTION

.NET Definition, The .NET strategy, the origins of the .NET technology, the .NET framework, benefits of the .NET approach, .NET Architecture-Common Language Runtime, MSIL, .NET Framework Classes. Advantages of Managed Code , The integrated Development Environment – menu bar, tool bar, from designer, project explorer, properties window, from layout window, Write, run, save, and print a project, Use online Help.

The form object: Properties, events and methods of forms;; Controls – Properties and events of different controls such as Text boxes, group boxes, check boxes, radio buttons, picture boxes and naming conventions. Variables, constants, Declaration of variables and constants considering data types and scope, Explicit data type conversions, Use formatting functions to format output. Predefined Dialogue Boxes, Input Box and Message Box.

UNIT II:

(10)

INTRODUCE CONTROLS AND THEIR PROPERTIES

The form object: Properties, events and methods of forms;; Controls – Properties and events of different controls such as Text boxes, group boxes, check boxes, radio buttons, picture boxes and naming conventions, Variables, constants and calculations: Declaration of variables and constants considering data types and scope, Explicit data type conversions, using arithmetic operators and order of operations, Use formatting functions to format output. Control Statements , Conditional statements and Looping statements. Menus, procedures and Functions. Toolbars and Status bar, Common Dialog (Open/SaveFile), ProgressBar

UNIT III:

(10)

DATABASE CONNECTIVITY

Data base basics & database engine, Create a database in Access Through VB, The nature of a relational databases, The data controls (DAO and ADO), Data Bound controls, Working with database objects in code, Data Manipulation through VB – Forms, Develop a database application. 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: Multiple Document Interface (MDI):Overview of MDI, Creating parent and child forms, Writing code for parent and child forms, Child window management, Creating applications.

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) @ Visual C#@ 2005: The Language by Donis Marshall
- The Complete Reference- Visual Basic .NET,Jeffrey R Shapiro,
- Microsoft Visual Basic .NET Programming for the Absolute Beginner, Jonathan S Harbour,2002
- Esposito Dino, Applied XML Programming for Microsoft .NET, Tata McGrawHill, 2003

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

TITLE : BASIC PROGRAMMING SKILLS

CODE : CSOE4218 (Open elective for other students)

Hrs / Week : 2 Hrs

CREDITS 2

Course Objectives:

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

- To provide an in-depth training for developing programming skills.
- To understand and develop programs independently.
- To understand the methods of debugging and correcting programs.
- To provide a proper foundation for learning other programming languages.

UNIT I (10)

INTRODUCTION TO PROGRAMMING

Problem Solving Using Computers: Language Classification, Problem Analysis, Algorithm and Flowchart design. Algorithms: Steps in developing algorithms, advantages and disadvantages. Flowcharts: Symbols used in developing flowcharts, advantages and disadvantages. Coding, testing and debugging. Documentation and maintenance. Program development and modular design.

INTRODUCTION TO C PROGRAMMING

History, Structure of a C program, C Conventions, Character Set, Identifiers, Keywords, Simple Data types, Modifiers, Variables, Constants, Operators (Arithmetic operator, relational operator, logical operator, ternary operator, unary operator, shorthand operator, bit-wise operator and arithmetic operator) Operator precedence. Input and Output operation: Single character input and output, formatted input and output, Buffered input.

UNIT II (10)

CONTROL STRUCTURES

Introduction, Conditional statement, if statement, if-else statement, nested if statement, else-if statement and switch statement. Go to statement. Looping statement, while statement, do-while statement, for statement, break and continue, nested for statement.

UNIT III (10)

ARRAYS

Introduction (One and two dimensional), Declaration of arrays, Initialization of arrays, processing with arrays. String manipulation, declaration of string arrays, string operations.

FUNCTIONS

Introduction, advantages of subprograms, Function definition, function call, Actual and formal arguments, local and global variables, function prototypes, types of functions, recursive functions, arrays and functions.

Self Study: Formatted Input output

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 : .NET TECHNOLOGIES LAB

CODE : CS4P1

Hrs / Week : 2 Hrs

CREDITS 1

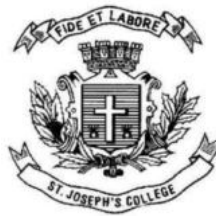
1. To Find the Roots of Quadratic Equation
2. To Create Electricity Bill with Graphical User Interface.
3. To Develop a Simple Calculator.
4. To Create a Student Interface with Graphical User Interface .
5. Design a VB application which has MDI and Child forms. Create a menu having the items such as file (New, Open),Format (Font, Regular, Bold ,Italic) and Exit in the MDI form.
6. Design an application to validate the user name and password and display message.
7. Design an application for Digital Clock.
8. To Implement Directory List Box, Drive List Box and File List Box .
9. Design an application to implement arithmetic operation using subroutine.
10. Design an application to implement scroll bar to the change the font size of the label and to create a Stop watch.
11. Design an application to create front end and back to implement ADO connection.
12. To Implement Animation and Graphics

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

Implemented for 2018 -19 Batch

Course Structure and Syllabus for

Bachelor of Science (Computer Science)
(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	CS5118	JAVA Programming	3	3	30	70	100
2	CS5218	Software Engineering	3	3	30	70	100
Practical							
1	CS5P1	JAVA Programming Lab	2	1	15	35	50
2	CS5P2	Web designing / Mini Project Lab	2	1	15	35	50

SIXTH SEMESTER:

Theory							
Sl. No.	Code	Course Title	Hrs / Week	Credits	IA	SE	Total
1	CS6118	Computer Networks	3	3	30	70	100

2	CS6218	Computer Graphics and Multimedia	3	3	30	70	100
Practical							
1	CS6P1	Major project lab	2	1	15	35	50
2	CS6P2	Computer graphics lab	2	1	15	35	50

FIFTH SEMESTER:

TITLE : JAVA PROGRAMMING.

PAPER CODE : CS 5118

NO OF HRS: 45HRS

CREDITS: 3

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:

Able to:

- Understand the basic concepts and principles of structured programming.
- Understand the basic concepts and principles of object oriented programming.
- Produce sample use-cases, pseudo-code, and an incremental coding plan for a given problem specification.
- Design, write, and test a Java program to implement a solution to a given problem specification.
- Understand the operation of common data structures and algorithms.

UNIT-1

INTRODUCTION TO JAVA:

10Hrs

JAVA Evolution, Introduction to Programming Languages, The Evolution of Java, Object-Oriented Programming Concepts and Java, The Primary Characteristics of Java, The Architecture, Simple Java Program, More of Java, An Application with Two Classes Java Program structure, Java Tokens, Java Statements, Implementing a Java Program, Java Virtual Machine, Programming Style.

Branching: Constants, Variables, and Using Data Types, Operators and Expressions, Type conversion and Associativity, Mathematical Functions. Decision Making and Introduction, Decision Making with if Statement, Simple if Statement, The if else Statement, Nesting of if else Statements, The else if Ladder, The Switch Statement, The?: Operator. Decision Making and Looping: Introduction. The while Statement, The do Statement, the for Statement, Jumps in Loops Labeled Loops.

Self- study: Java History, Differences between C++ and Java,

UNIT-2

CLASSES, ARRAYS, STRINGS AND COLLECTION FRAME WORK: 10Hrs

Introduction, Defining a Class, Adding Variables, Adding Methods, Creating Objects, Accessing Class Members, Constructors, Methods Overloading, Static Members, Nesting of Methods.

Inheritance: Extending a Class Overriding Methods, Final Variables and Methods, Finalized methods, Abstract Methods and Classes, Visibility Control.

Arrays Strings and Vectors: Arrays, One – dimensional Arrays, Creating an Array, Two – dimensional Arrays, Strings, Vectors and Wrapper Classes.

UNIT-3

INTERFACES, PACKAGES, AND MULTITHREADED PROGRAMMING: 6 Hrs

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.

Self-study : implementing of Interface, difference classes.

UNIT -4

MULTITHREADED PROGRAMMING:

10 Hrs

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.

Self study: thread class example, synchronization strategies

UNIT-5

MANAGING EXCEPTIONS, APPLLET PROGRAMMING:

10 Hrs

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

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, More about Applet Tag, Passing Parameters to Applets, Aligning the Display, More About HTML Tags, Displaying Numerical Values, Getting Input from the User.

TEXT BOOK :

1. "Introduction to Java Programming" by Daniel Liang.

2. E. Balaguruswamy, Programming with JAVA, A Primer, TMH (1999)

REFERENCES:

1. Darrel Ince & Adam Freeman, Programming the Internet with Java, Addison – Wesley, (1997).
2. KenArnold & James Gosling, The Java Programming Language, Addison – Wesley, (1998)
3. Patrick Naughton & Herbert Schildt, JAVA 2: The Complete Reference, 3rd Edition, TMH, (1999). (1)

Units	2 marks(10 questions)	6 marks(7 questions)	10 marks(3 questions)	Total
1	2	2	-	16
2	2	2	-	16
3	2	1	1	20
4	2	1	1	20
5	2	1	1	20

TITLE : **Software Engineering**
CODE : **CS5218**
Hrs / Week : **3 Hrs**
CREDITS : **3**

Course Objective

- To inculcate in students’ different concepts of software engineering principles
- To develop the skills necessary to design, develop and execute software projects.
- 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.
- Understand various aspects of design, coding, testing and reusability in software development.

UNIT 1

5 Hrs

Introduction to software engineering, Software Products and Software process, Process models: Waterfall model, Spiral model, prototyping model- evolutionary and throwaway prototyping models, incremental delivery, Agile methods. Software development life cycle.

UNIT 2

10 Hrs

The requirement Engineering Process, Requirement analysis- view point oriented and method-based analysis, System models -dataflow model, semantic model and object model, Functional and Nonfunctional requirement, software requirement document, software cost factors, Cost estimation techniques, COCOMO model.

UNIT 3

10 Hrs

Design Process, Design Strategies- functional and object-oriented designs, Design Quality- cohesion, coupling, Architectural design -System Structuring, Control modelling, Modular decomposition, Domain Specific architecture. Structured Coding Techniques – Coding Style – Standards and Guidelines – Documentation Guidelines.

UNIT 4

10Hrs

Software reliability metrics, Software reliability Specification, Statistical testing, , Fault avoidance & tolerance, Exception handling, Software development with reuse, Software development for reuse, Generator based reuse..

UNIT 5

10 Hrs

The testing Process, Test Planning & Strategies, Black Box, Structural, interface testing, Program inspections, Mathematically based verification, Static analysis tools, Clean room software development.

Self-study: software project management- introduction, types of software projects, Overview of project planning in stages.

BOOKS :

1. Ian Sommerville, Software Engineering, 6th Edition, Pearson Publication Ltd. 2001

2. Roger Pressman, Software Engineering – A practitioner’s approach (McGraw Hill).
3. Pankaj Jalote, An Integrated Approach to Software Engineering – Narosa Publishing house.
4. Bob Hughes, Mike Cotterell , Rajib mall , Software project management , Mc Graw hill Education publication.

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

TITLE: JAVA PROGRAMMING LAB
PAPER CODE: CS5P1
CREDIT: 1

1. Write a Java program that prints all real solutions to the quadratic equation $ax^2 + bx + c=0$. Read in a, b, c and use the quadratic formula. If the discriminate $b^2 - 4ac$ is negative, display a message stating that there are no real solution .
2. To accept 5 subject marks through command line arguments. Find the average and total marks display the result in various grades as follows.
3. Write a java program to calculate gross salary & net salary taking the following data Input: empno, empname , basic Process: DA=50%of basic HRA=25%of basic CCA=Rs240/- PF=10%of basic PT=Rs100/-
4. Create a single dimensional array type of string and display the text in alphabetical order.
5. Write a java program that implements educational hierarchy using multi level inheritance.
6. Write a program to implement constructor overloading by passing different number of parameter of different types
7. Write a Java program to find area of geometrical figure using the method overloading
8. Generate a java program to show the usage of method overriding.
9. Generate a java program which shows the difference between static, final and abstract keywords.

10. To generate the list of string operation (any 7 function) using buffer reader.
11. Create an interface with all arithmetic operations and implement it to demonstrate interface implementation.
12. Write a program to identify the accessibility of a variable by means of different access specifier within and outside package
13. Generate a single thread a) using thread class b) using Runnable class
14. Write a program to implement thread priorities
15. Create a java program to demonstrate the usage of nested try block statement.
16. Write a program to sort list of elements in ascending and descending order and show the exception handling.
17. Write a program to create student report using applet.

Scheme of valuation:

Internals

Average of all the Eleven Labs, Each lab carries 10 marks 15 Marks.

External

- | | |
|---|----------|
| 1. Writing two programs one from each section | 10 marks |
| 2. Execution of one program | 5 marks |
| 3. Formatting the program and output | 05 marks |
| 3. Record verification | 05 marks |
| 4. Viva voce related to practical topics only | 10 marks |

External Total 35 marks

Total Marks 50

TITLE: WEB DESIGNING / MINI PROJECT LAB

PAPER CODE: CS5P2

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 (web based project)
- 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.net, C#.Net, Java, Asp.net

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	AIRLINE RESERVATION SYSTEM
9	AIRLINE TICKET BOOKING SYSTEM
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 Valuation

- | | |
|---|----|
| 1. Average for all Eleven labs , each lab carries 10 marks
marks | 10 |
| 2. Verification of Report with Explanation
marks | 5 |

Total Internal

15 Marks

External Valuation

- | | |
|-----------------------------------|----------|
| 1. Demonstration and presentation | 25 marks |
| 2. Documentation | 10 marks |

Total External

35 Marks

Total Marks

50 Marks

SIXTH SEMESTER:

TITLE: COMPUTER NETWORKS

PAPER CODE: CS6118

TOTAL TEACHING HOURS: 45

CREDITS: 03

Course Objective:

- To study about network components.
- To study about topologies and network models.

Learning Outcome:

- Students will gain knowledge about networks, internal components and its functionality.
- To study about various network protocols, algorithms.

UNIT – I

INTRODUCTION:

8 Hours

Communication Network and services, Approaches to Network Design, Network Functions, Message ,packet and circuit Switching , Internet, Packet Switching ; Key factors in Communication Network Evolution ; Layered Architecture and Applications – Examples of Layering , OSI Reference Model, TCP/IP Model Telnet FTP and IP Utilities.

Self study: Network Topology, applications of networks.

UNIT – II

10 Hours

TRANSMISSION SYSTEMS:

Digital transmission Systems – Twisted Pair , Coaxial Cable, Optical Fibre, Radio Transmission Infrared Light, Multiplexing – frequency – Division , Time – Division , Wavelength Division Multiplexing, Switching- Circuit , Packet and Message.

Error detection, Two – dimensional parity checks, Internet checksum, Polynomial code; standardized Polynomial codes, Error detecting capability of a polynomial code.

Self- study: Properties of media

UNIT

-

III**10 Hours****PEER –TO-PEER PROTOCOLS:**

Peer-to peer Protocols and service models ARQ Protocols stop and wait , Go –back-N Selective Repeat , Transmission efficiency of ARQ Protocols, Other adaptation functions , - Sliding window flow control Timing Recovery in Synchronous Services Reliable Stream Service, Data Link Control.

Self- study: examples of ARQ.

UNIT - IV**10 Hours****LOCAL AREA NETWORKS AND MEDIUM ACCESS CONTROL PROTOCOLS:-**

Multiple access communications; Local Area network – LAN Structure, MAC Sub layer, Logical link control layer, Random Access protocols ALOHA , Slotted ALOHA, CSMA, CSMA/CD, Scheduling approaches to medium access control – Reservation Systems, polling , Token passing rings, Channelization – FDMA, TDMA, CDMA.

Self- study: examples of FDMA, TDMA, CDMA

UNIT

-

V**9 Hours**

LAN Standard –Ethernet and IEEE, 802.3 LAN Standard ; Token Ring and IEEE 8025 LAN standard , FDDI, Wireless LAN's and IEEE 802.11 Standards; Routing algorithms classification , Routing tables, shortest path routing algorithms, Flooding , Hierarchical routing , Distance vector routing Link state routing, congestion control algorithms.

Self- study: examples of Hierarchical routing, Distance vector routing Link state routing, congestion control algorithms.

Text Book:

1. Stallings, “Data and Computer Communications”, 7th Edition, Pearson Education, 2012

Reference:

1. Andrew S Tanenbaum, “Computer Networks”, 4th Edition, Pearson Education.

2. Behrouz Forouzan, Introduction to Data Communication & Networking TMH, 1999.

3. Larry & Peterson & Bruce S Davis; Computer networks Second Edition, Morgan Kaufman, 2000.

UNITS	2 MARKS	6 MARKS	10 MARKS	TOTAL
UNIT 1	2	1	-	10
UNIT 2	2	1	-	10
UNIT 3	2	1	1	20
UNIT 4	2	2	1	26
UNIT 5	2	2	1	26

TOTAL	20	42	30	92
-------	----	----	----	----

PAPER TITLE: COMPUTER GRAPHICS AND MULTIMEDIA

PAPER CODE: CS 6218

Total Teaching Hours for Semester: 45

No of Lecture Hours/Week: 3

Objectives:

- To study the graphics techniques and algorithms.
- To study the multimedia concepts and various I/O technologies.
- To be familiar with various clipping techniques
- To enable the students to improve their creativity

Learning Outcome:

- To create image that can translate, scale and rotate.
- To create repeatable images for use as backgrounds, texture etc. Evaluate graphic projects, identify items for improvements and implement changes.
- To write programs on Compression and Decompression algorithm.
- Knowledge on various formats of data and information representation.
- To function as designers, using tools for analyzing the world, accessing and interpreting information and representing what they know to others.

Unit-1: OUTPUT PRIMITIVES AND DISPLAY DEVICES

10 hours

Introduction: Points and Line, Line Drawing Algorithms-DDA and Bresenham's line algorithm, Circle drawing Algorithms- DDA and Mid-Point algorithm, Polygon filling algorithms, Attributes of output primitives. Display devices-Cathode Ray Tube, LCD and LED Techniques

Unit-2: TWO-DIMENSIONAL CONCEPTS

10 hours

Two-Dimensional Geometric Transformations – Basic Transformations- Translation, Rotation, Scaling. Other Transformations-Shearing and Reflection. Composite Transformation. Two-Dimensional Clipping and Viewing. Cohen Sutherland Line clipping algorithm.

Unit-3: THREE-DIMENSIONAL CONCEPTS**10 hours**

Three-Dimensional Geometric Transformations – Basic Transformations in 3D-Translation, Rotation, Scaling. Other Transformations-Shearing and Reflection. Three-Dimensional Viewing-Parallel and Perspective Projections. Polygon Surfaces, Curved Surfaces, Quadtree and Octrees.

Unit 4: MULTIMEDIA INFORMATION REPRESENTATION**5 hours**

Introduction to multimedia representation, Text-Unformatted, Formatted and Hypertext. Images-Graphics, Digitized document and Digitized pictures. Audio-PCM speech and CD quality audio.

Unit 5: INTRODUCTION TO MULTIMEDIA TECHNOLOGIES**10 hours**

Multimedia applications, Multimedia System Architecture, Evolving technologies for Multimedia, Multimedia Data interface standards, Multimedia Databases. Compression & Decompression- lossy and lossless compression.

Self-Study: Examples of lossy and lossless compression

TEXT BOOKS

1. Donald Hearn and Pauline Baker, "Computer Graphics C Version", Pearson Education.
2. "Multimedia Communications: Applications, Networks, Protocols and Standards" By Halsall, Fred, Pearson Education.
3. Prabat K Andleigh and Kiran Thakrar, "Multimedia Systems and Design", PHI.

REFERENCES

1. Judith Jeffcoate, "Multimedia in practice technology and Applications", Pearson.
2. Foley, Vandam, Feiner, Huges, "Computer Graphics: Principles & Practice", Pearson Education.
3. Principles of Interactive Computer Graphics-Newman and Sproull, McGraw Hill.
4. Cooley: The Essence of Computer Graphics, Pearson Education.

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

TITLE: MAJOR PROJECT LAB

PAPER CODE: CS6P1

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 Marks 30**External**

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

**Total External Marks
70****Total Marks 100
Marks****TITLE: COMPUTER GRAPHICS LAB
PAPER CODE: CS6P2
CREDIT: 1****LAB PROGRAMS**

1. Study the basic graphics function defined in "Graphics.h".
2. Write a program to draw a hut or another geometrical figures.
3. Write a program to draw a line using DDA algorithm.
4. Write a program to draw a line using Bresenham's algorithm.
5. Write a program to draw a circle using DDA algorithm.
6. Write a program to draw a circle using Bresenham's algorithm.
7. Write a program to draw an ellipse using mid-point algorithm
8. Write a program to draw a triangle and translate it.
9. Write a program to draw a line and increase it length three times.
10. Write a program to draw a square and rotate it by 45 degrees.
11. Write a menu driven program to draw a triangle and reflect it along the following axis:
 - a. Along x-axis
 - b. Along y-axis
 - c. Along $x = y$ axis

12. Write a program to perform Shearing of a square along both x and y direction.
 13. Write a program to perform Composite translation.
 14. Write a program to implement polygon filling.
 15. Write a program to draw a moving car.
-

Scheme of valuation:

Internals

Average of all the Eleven Labs, Each lab carries 10 marks 15 Marks.

External

1. Writing two programs one from each section 10 marks
2. Execution of one program 5 marks
3. Formatting the program and output 05 marks
3. Record verification 05 marks
4. Viva voce related to practical topics only 10 marks

External Total 35 marks

Total Marks : 50