Course Structure and Syllabus for

Bachelor of Computer Applications (BCA)

(Six Semester Course)

Proposed for 2015 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 programme. 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

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	CA1116	Computer Fundamentals And Problem Solving	4	4	30	70	100
4	CA1216	Mathematical Foundation For Computer Science	4	4	30	70	100
5	CA1316	Programming In C	4	4	30	70	100
6	CA1416	Digital Electronics	4	4	30	70	100
Practical							
1	CA1P1	C Programming Lab	2	1	70	30	100
2	CA1P2	Automation 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	CA2115	Data Structures Using C	4	4	30	70	100
4	CA2215	Microprocessors	4	4	30	70	100
5	CA2315	Operating System	4	4	30	70	100
6	CA2415	Computer Oriented Numerical Analysis	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	CA3115	OOPS Using C++	4	4	30	70	100
4	CA3215	Visual Programming	4	4	30	70	100
5	CA3315	Database Management Systems	4	4	30	70	100
6	CA3415	Software Engineering	4	4	30	70	100
Practical							
1	CA3P1	C++ Programming Lab	2	1	70	30	100
2	CA3P2	Visual Programming 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	CA4115	Computer Graphics	4	4	30	70	100
4	CA4215	Computer Networks – I	4	4	30	70	100
5	CA4315	Java Programming	4	4	30	70	100
Open Electives (For students of other courses)							
6	CAOE4415	Web Development	2	2	15	35	50
7	CAOE4515	Basic Programming Skills	2	2	15	35	50
Practical							
1	CA4P1	Java Programming Lab	2	1	70	30	100
2	CA4P2	Computer Graphics Lab	2	1	70	30	100

FIFTH SEMESTER

Theory							
Sl.No.	**Code**	**Course Title**	**Hrs / Week**	**Credits**	**IA**	**SE**	**Total**
1	CA5115	Computer Organization And Architecture	4	4	30	70	100
2	CA5215	Multimedia Technology	4	4	30	70	100
3	CA5315	Web Technologies	4	4	30	70	100
4	CA5415	Computer Network-II	4	4	30	70	100
Practical							
1	CA5P1	Web Technologies 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	CA6115	E-Commerce	4	4	30	70	100
2	CA6215	Object Oriented System Design	4	4	30	70	100
3	CA6315	.NET Programming	4	4	30	70	100
4	CA6415	Mobile Applications	4	4	30	70	100
Practical							
1	CA6P1	.NET Programming Lab	2	1	70	30	100
2	CA6P2	Major Project 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.

FIRST SEMESTER

TITLE : Computer Fundamentals and Problem Solving
CODE : CA1116
Hrs / Week : 4 Hrs
CREDITS : 4

Course 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 designing

Learning Outcome:

After learning the course the students should be able to explain about the fundamentals of computers. The student should be able to solve any types of problem. The students should be able to explain the working of the CPU and other components of the motherboard. The students should be able to use simple software.

UNIT A FUNDAMENTALS OF COMPUTERS 8 HRS

Input, Output and Memory devices 5 Hrs

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.

Software Concepts 3 Hrs

Introduction, Types of Software (Application and System software), Introduction to Operating Systems, Functions of Operating Systems, Types of Operating Systems,
Functional features of commonly used operating systems

UNIT B CONFIGURING A COMPUTER 12 HRS

Computer components 8 Hrs

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, Power consumption of PC.

Advanced features 4 Hrs

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.

UNIT C PROBLEM SOLVING METHODOLOGY 15 HRS

Problem Solving Methodology 1 Hrs

Introduction to problem Solving, problem definition and problem Analysis

Design Of a Solution 8 Hrs

Algorithms and Flowcharts

Development of Programs (Coding, testing, debugging) 1 Hrs

Documentation and Maintenance

Programming Constructs 2 Hrs

Sequence, Selection and Iteration

Characteristics of a Good Program 1 Hrs

Types of Errors

Approaches to Problem Solving 2 Hrs

Top-down, Bottom-up, Modular, Structured and object oriented

UNIT D ELEMENTARY CONCEPTS OF WORD PROCESSING, SPREADSHEETS AND WEB DESIGNING 25 HRS

Word Processing 8 Hrs

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

Spreadsheets 12 Hrs

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).

Web Designing 5 Hrs

Introduction to the Internet, Services on the Internet, Some Definitions related to the web

BOOKS:

1. Fundamentals of Computers by Rajaraman, PHI, 1986, 2nd Edition.
2. Computer Organization by Carl Hamacher V. Zaki, McGraw Hill, 1990
3. Digital computer Fundaments by Malvino & Leach.
4. Office Automation and word processing – E Balaguruswamy
5. Dummies Series – MS-Office
6. M. Ercegovac, T. Lang, J.H. Moreno, **Introduction to Digital Systems**, John Wiley and Sons

TITLE : Mathematical Foundation For Computer Science
CODE : CA1216
Hrs / Week : 4 Hrs
CREDITS : 4

Course Objective

- To inculcate in students the fundamental mathematical background in computer science.
- To gain knowledge about Sets, Relations Functions, Matrices, Mathematical logic, and Group theory.

Learning Outcome

Upon successful completion of this course student should be able to

- Understand the basic concepts of Sets, Relations Functions, Matrices, Mathematical logic, and Group theory.
- Develop analytical ability to solve real-world problems using these methodologies.

Set Theory **25 Hrs**

Sets, Relations and Functions Definition of set, intersection, union and compliments. Principal of inclusion and exclusion, De Morgan's laws, cardinality difference, symmetric difference. Cartesian Product, relations. **7 Hrs**

Matrix Theory Review of fundamentals, equivalent matrices, elementary row (column) operations, rank of a matrix by reducing it to the normal form, rank of a matrix by reducing it to echelon form. **8 Hrs**

Mathematical Logic Connectives, Negation, Conjunction, Disjunction, conditional, bi-conditional, statement formulas, Tautology and contradiction, Equivalence formulae Normal forms: Principle conjunctive and disjunctive normal forms, Theory of inferences for statement calculus validating using truth tables. **10 Hrs**

Graph Theory **35 Hrs**

Graph Theory: Definition of a Graph, Finite and Infinite Graphs, Incidence and Degree of a vertex, Null Graph, Sub graphs, Walks, Paths, Circuits, Connected, Disconnected graphs and Components, Euler Graph, Hamiltonian Path and Hamiltonian Circuits. **10 Hrs**

Trees And Matrix Representation: Properties of Trees, Distance and Centres in a Tree, Rooted and Binary Trees, Spanning Trees and Fundamental Circuits. Cutset, properties of a Cutset. Matrix Representation of graphs: Incidence matrix, Circuit matrix, Fundamental Circuit matrix, Cutset matrix, Path matrix, Adjacency matrix **10 Hrs**

Planar and Dual Graphs Planar Graphs, Kurtowski's two Graphs, Different Representations of a Planar Graph, Detection of Planarity. **7 Hrs**

Directed Graphs: Definition, Some types of Digraphs, Digraphs and Binary relations, Directed paths and Connectedness, Euler Digraphs, Trees with directed edges, Fundamental Circuits in Digraphs, Adjacency Matrix of a Digraph. **8 Hrs**

BOOKS:

1. Engineering Mathematics by H.C. Das, Chand publications.
2. Graph theory – Narasingh Deo
3. Discrete mathematical Structures by J.P. Trembley and R. Manohar, TMH Publications.
4. Discrete Mathematics by Liu.
5. BCA, Mathematics Vol-II G.K. Ranganath and B. Soorya Narayana.

TITLE : Programming In C
CODE : CA1316
Hrs / Week : 4 Hrs
CREDITS : 4

Course Objective

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.

Learning Outcome

- To analyze problems efficiently and develop comprehensive logic to solve it.
- To develop good algorithms and flowcharts to solve problems.
- To write C programs in a structured manner.

Introduction To C Programming **5 Hrs**

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.

Control Structures **10 Hrs**

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.

Arrays **9 Hrs**

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

Functions **10 Hrs**

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.

Storage Classes, Structures and Unions **5 Hrs**

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 **10 Hrs**

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

Files **6 Hrs**

Introduction, File data type, opening and closing a file, file functions (getc, putc, getw, putw, fscanf, fprintf, fread, fwrite, fgets, fputs, feof).

BOOKS:

1. Kanetkar, Yashavant: "**Let Us C**", 4th Edition. BPB Publications.
2. Gottfried, Byron S: "**Programming with C",** 1996. Tata McGraw-Hill
3. Balagurusamy, E: "**Programming in ANSI C**" 2nd Edition. Tata McGraw-Hill
4. Deitel, H M and Deitel P J: "**C How to Program**", 2nd Edition. Prentice-Hall.

TITLE : Digital Electronics
CODE : CA1416
Hrs / Week : 4 Hrs
CREDITS : 4

Course Objectives:

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

Learning Outcome:

After learning the course the students should be able to explain about the fundamentals of computers, digital number systems and logic circuits. The student should be able to solve logic function minimization. The students should be able to differentiate between combinational and sequential circuits such as decoders, encoders, multiplexers, de-multiplexers, flip-flops, counters, registers. The students should be able state the specifications of logic families. The student should be able to explain the different types of computer memories

Introduction to computers **8 Hrs**

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.

Representation of Data **15 Hrs**

Number Systems-binary ,octal, hexadecimal,, and Inter-conversions among them, Binary arithmetic-Addition, Subtraction, multiplication, division, Binary number system complements-9's,10's compliments (additions only) 1's and 2's complements and their subtractions, computer codes-ASCII, Excess- 3 code and Gray code, EBCDIC code and BCD code. Fixed point representation of numbers, floating-point representation

Boolean Algebra and Logic Circuits **12 Hrs**

Boolean Algebra Laws and theorems, Gates- AND, OR, NOT,NAND, NOR, EXOR, truth tables, Boolean expressions and their simplifications, SOP & POS-min-term, max-term & Karnaugh map simplification of logic expression.

Combinational and Sequential circuits **15 Hrs**

Multiplexers, De-multiplexers, Decoders, Encoders, Half Adder, Full Adder, Parallel Adder-subtractor, Flip flops- RS, JK, D, T, Master Slave , Counters-ripple,mod-3,mod-5,mod- 10,ring counter, Shift registers.

Memory **10 Hrs**

Need for memory devices, computer memory devices, primary and secondary Memory hierarchy, ROM, RAM Types of semiconductor memories:- ROM, PROM, EPROM, EEPROM, Flash, RAM, SRAM, DRAM ,Cache Memory

BOOKS:

1. Digital Computer Fundaments by Thomas C Bartee, McGraw Hill, VI Edition, 1987
2. Computer Organization by Carl Hamacher V. Zaki, McGraw Hill, 1990
3. Digital computer Fundaments by Malvino & Leach.
4. Digital Computer Fundaments by Malvino.
5. Malvino, **Digital Principles and Applications**, Tata McGraw Hill, 4th Edition
6. M. Ercegovac, T. Lang, J.H. Moreno, **Introduction to Digital Systems**, John Wiley and Sons

PRACTICALS

TITLE : C PROGRAMMING LAB
CODE : CA1P1
Hrs / Week : 2 Hrs
CREDITS : 1

Guidelines:

. Follow Coding standards and write structured programs.
. The output of the programs should be neatly formatted.
. The programs should be user friendly and interactive.
. Use comments wherever necessary.

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

8. To demonstrate the concept of File

a. Write a program to create a file and store some records in it. Display the contents of the same.

TITLE : PC – PACKAGE LAB
CODE : CA1P2
Hrs / Week : 2 Hrs
CREDITS : 1

1. Commands in DOS(Internal and External) and simple exercises on TYPE, REN, DEL, CD, MD, COPY, TREE, BACKUP commands

2. Exercises on entering text and data (Typing Practice)

3. **Features of Windows as an operating system**

- Start, Shutdown and restore
- Creating and operating on the icons
- Opening, closing and sizing the windows
- Using elementary job commands like
- Creating, saving, modifying,
- Renaming, finding and deleting a file
- Creating and operating on a folder
- Changing setting like, date, time, color (back ground and fore ground)
- Using short cuts
- Using on-line help

4. MS-Word

File Management:

Opening, creating and saving a document, locating files, copying contents in some different file(s), protecting files, Giving password protection for a file.

Page Set up:

Setting margins, tab setting, ruler, indenting

Editing a document:

Entering text, Cut, copy, paste using tool-bars

Formatting a document:

Using different fonts, changing font size and colour, changing the appearance through bold/ italic/ underlined, highlighting a text, changing case, using subscript and superscript, using different underline methods

- Aligning of text in a document, justification of document ,Inserting bullets and numbering
- Formatting paragraph, inserting page breaks and column breaks
- Use of headers, footers: Inserting footnote, end note, use of comments
- Inserting date, time, special symbols, importing graphic images, drawing tools

Tables and Borders:

Creating a table, formatting cells, use of different border styles, shading in tables, merging of cells, partition of cells, inserting and deleting a row in a table

- Print preview, zoom, page set up, printing options
- Using Find, Replace options

Using Tools like:

Spell check, help, use of macros, mail merge, thesaurus word content and statistics, printing envelops and lables

- Using shapes and drawing toolbar,
- Working with more than one window in MS Word,
- How to change the version of the document from one window OS to another
- Conversion between different text editors, software and MS word

5. MS-Excel

Starting excel, open worksheet, enter, edit, data, formulas to calculate values, format data, create chart, printing chart, save worksheet, switching from another spread sheet.

Menu commands: Create, format charts, organize, manage data, solving problem by analyzing data, exchange with other applications. Programming with MS-Excel, getting information while working.

Work books: Managing workbooks (create, open, close, save), working in work books, selecting the cells, choosing commands, data entry techniques, formula creation and links, controlling calculations, working with arrays

- Editing a worksheet, copying, moving cells, pasting, inserting, deletion cells, rows, columns, find and replace text, numbers of cells, formatting worksheet.

Creating a chart: working with chart types, changing data in chart, formatting a chart, use chart to analyze data

- Using a list to organize data, sorting and filtering data in list
- Retrieve data with MS – query: Create a pivot table, customising a pivot table.

Statistical analysis of data

Customise MS-Excel:

How to change view of worksheet, outlining a worksheet, customize workspace, using templates to create default workbooks, protecting work book

- Exchange data with other application: linking and embedding, embedding objects, linking to other applications, import, export document.

6. MS PowerPoint

Creating slides in power point.

SECOND SEMESTER

TITLE : Data Structures Using C
CODE : CA2115
Hrs / Week : 4 Hrs
CREDITS : 4

Course Objective

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.
Objectives of the course are:

- To be able to practically implement the data structures like stack, queue, array etc.
- To understand and implement different searching and sorting

Learning Outcome

- Understand the need for Data Structures when building Applications.
- Appreciate the need for optimized algorithm.
- Able to walk through insert and delete for different data techniques.
- Improve programming skills.

Introduction to data structures: 8 Hrs
Introduction. Classification of data structures. Analysis of algorithms, space and time complexity, best and worst case, asymptotic notation, upper and lower bounds.

Linear data structures: 15 Hrs
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.

Linked Lists: 15 Hrs
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.

Trees: 10 Hrs
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.

Searching and sorting: 12 Hrs
Sorting – Bubble sort , Insertion sort, selection sort, quick sort, merge sort.
Searching – Linear and Binary.

BOOKS:

1. Data Structure by Schaum Series
2. Fundamentals of Data Structure by Horowitz Sahni
3. Data Structure by Dale and Lilly
4. S. Sahni, **Data Structures**, **Algorithms and Applications**, Tata McGraw Hill,

TITLE : Microprocessors
CODE : CA2215
Hrs / Week : 4 Hrs
CREDITS : 4

Course Objective

To help students to understand the basics of 8085 microprocessor-based Systems and assembly language programming.

Learning outcome

At the end of this subject, students should be able to :

- Identify the basic element and functions of 8085 microprocessor.
- Describe the architecture of 8085 microprocessor.
- Apply the programming techniques in developing the assembly language program.

Architecture and Operation: 8 Hrs

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

Programming the 8085: 20 Hrs

Programming model, instruction classification, Instruction format, Addressing modes, writing assembly level programs-overview of instruction set, timing diagrams. Data transfer, Arithmetic, Logic branch operations. Programming techniques - Looping, Counting and Indexing, Additional data transfer and 16 bit arithmetic operations, logic operations, Compare and rotate operations. Counters and Time delays. Stacks and subroutines-conditional CALL and RETURN instructions. Advanced subroutine concepts. Code Conversion- BCD to Binary and Binary to BCD conversions, BCD to 7 segment conversion, BCD addition and subtraction, multiplication and division.

Memory Interfacing ,Interfacing I/O Devices: 7 Hrs

Memory Interfacing, Memory interfacing circuit. Interfacing Output displays, Interfacing Input devices, peripheral mapped I/O, memory mapped I/O.

Interrupts: 7 Hrs

8085 interrupts,8085 vectored interrupts, Restart as Software instructions, Additional I/O concepts and processes**.**

Interfacing of peripherals (I / Os) and applications: **18 Hrs**

Interfacing Keyboard (linear and matrix) and 7 segment display including multiplexers, 8279 programmable keyboard/display interface, 8255 PPI, 8259 PIC, DMA and 8257 DMA controller, Interfacing data converters-D to A converters, A to D converters.

BOOKS:

1. R.S. Gaonkar - Microprocessor Architecture, Programming and Application with 8085. Penram Int. 3rd Edn.
2. Brey B Brey, The Intel Microprocessor 8086/8088, 80186/80188, 80286, 80386, 80486 Pentium, Pentium Pro. Processor, 6th Edition, Pearson Education.
3. Douglas V. Hall - Microprocessors and digital systems, MH.
4. Kenneth L. Short - Microprocessor and Programmed Logic", PHI, 2nd Edn.
5. Aditya P. Mathur - Introduction to Microprocessors, 3rd Edn. TMH
6. Antonakos: Introduction to Intel family of Microprocessors Pearson Education
7. Hoffer: Modern Systems Analysis and Design Pearson Education
8. Kendall, System Analysis and Design

TITLE : Operating system
CODE : CA2315
Hrs / Week : 4 Hrs
CREDITS : 4

Course Objective

- To introduce to 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

Upon completion of the course students will be able to:

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

Introduction: **10 Hrs**

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

Process Management: **15 Hrs**

Process concept, functions, Cooperating process, Interprocess communication, Scheduling Criteria, job & processor scheduling. (preemptive & non preemptive) FCFS,SJF,Round Robin algorithms, process synchronization – semaphores, deadlocks - detection and recovery.

Memory Management: **15 Hrs**
Functions, Different schemes-Single continuous-Partitioned -Multiple Relocatable-Paging-Demand paging-Segmentation-Paged segmentation. Virtual Memory management: Demand paging, Page replacement & page replacement algorithm, and thrashing.

Device and file system: **10 Hrs**
Disk structure, allocation methods, free space management, need for disk scheduling, scheduling algorithm, access methods, directory structure, organization, file protections.

Case Studies: **10 Hrs**
DOS and UNIX– Memory Management – overlaying – Extended and Expanded memory – Memory allocation – File system and allocation method – Internal and External commands – Memory management commands – File management commands.

BOOKS:

1. Operating System by Milan Milenkovic, McGraw Hill, 1987
2. Operating System by Madnick and Donoval, McGraw Hill, 1974
3. Operating System Concepts by James L Peterson
4. Operating System Design and Implementation by Andrew S Tenenbaum

TITLE : Computer Oriented Numerical Methods
CODE : CA2415
Hrs / Week : 4 Hrs
CREDITS : 4

Course Objective:

The objective of this course is to enable 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. They would also gain an appreciation of the concept of error in these methods and need to analyze and predict it.

Learning Outcomes

- Be aware of the use of numerical methods in modern scientific computing,
- Be familiar with finite precision computation,
- Be familiar with calculation and interpretation of errors in numerical methods,

Computer Arithmetic and Solution of Non-Linear Equations: **10 Hrs**
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.

Solution of Non-Linear equations: **12 Hrs**
Iterative method, Successive Bisection– Fixed point – Regula falsi – 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.
12 Hrs

Interpolation and Curve Fitting: 14 Hrs
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.

Numerical differentiation and Integration : 12 Hrs
Differentiation based on polynomial fit - Numerical integration using Simpson's rule and Gaussian quadrature formula - Numerical solution of differential equations of the form dy/dx=f(x,y) using Euler's method .

BOOKS:

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

PRACTICALS

TITLE : DATA STRUCTURES LAB
CODE : CA2P1
Hrs / Week : 2 Hrs
CREDITS : 1

Guidelines:
. Follow Coding standards and write structured programs.
. The output of the programs should be neatly formatted.
. The programs should be user friendly and interactive.
. Use comments wherever necessary.

1. Illustrate function using 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. Evaluate an infix expression
18. 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

1. Exchange of two 16-bit numbers.
2. Addition & 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 &16 bit number.
9. Largest & smallest number in an Array
10. Checking of +ve, -ve &Zero.
11. Multiplication by successive addition.
12. Frequency of occurrence of element.
13. Sum & Average of n-one byte number
14. Square root of a given number.
15. Searching for a number.
16. Addition of two 32 bit numbers.
17. Counting no of ones and zeroes.
18. Sum of odd and even numbers in an array.

THIRD SEMESTER

TITLE : OOPS Using C++
CODE : CA3115
Hrs / Week : 4 Hrs
CREDITS : 4

Course Objective

- To familiarize with Object Oriented concepts
- To develop the skills of programming in C++
- To understand the difference between object oriented and procedure oriented programming.

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

Introduction: 2 Hrs

Procedural Languages, definition of OOP, Basic concept of OOP, Object, Class, Data Abstraction, Data Encapsulation, Data Hiding member functions, Reusability, Inheritance, Creating new Data Types, Polymorphism, Overloading, Dynamic binding, and Message passing.

C++ Features: 2 Hrs

The iostream class, C++ Comments, C++ Keywords, Variable declaration, The Const Qualifier. The Endl, Set W, set precision, Manipulators, The scope resolution operator, new & delete Operators.

Functions: 8 Hrs

Simple Functions: Function declaration, calling the function, function definition; Passing argument to, returning value from function; passing constants, Variables, pass by value, passing structure variables, pass by reference, Default arguments, return statements, return by reference, overloaded functions; Different number of arguments, Different Kinds of arguments, inline function.

Objects and Classes: 12 Hrs

Classes & Objects, Class Declaration, Class members; Data Constructors, Destructors, Member functions, Class member visibility; private, public, protected. The scope of the class object constructors; Default Constructor, Constructor with argument, constructor with default arguments, Dynamic constructor, copy constructor, Overloaded constructor, Objects as function arguments; member functions defined outside the class, Objects as arguments, returning objects from functions, class conversion, manipulating private Data members, Destructors, classes, objects & memory, array as class member data, Array of objects, string as class member.

Operator Overloading: **12 Hrs**

Overloading unary operator: Operator Keyword, Operator Arguments, Operator return value, Nameless temporary objects, limitations of increment operator, overloading binary operator, arithmetic operators, comparison operator, arithmetic assignment operator, Data conversion; conversion between Basic types, Conversion between objects & Basic types, conversion between objects of different classes.

Inheritance: **12 Hrs**

Derived Class & Base Class: Specifying the Derived class accessing Base class members, the protected access specifier, Derived class constructor, Overriding member functions, public and private inheritance; Access Combinations, Classes & Structures, Access Specifiers, Level of inheritance; Multilevel inheritance, Hybrid inheritance, Multiple inheritance; member functions in multiple inheritance, constructors in multiple inheritance.

Virtual Functions: **6 Hrs**

Normal member function accessed with pointers, Virtual member functions accessed with pointers, pure virtual functions, Friend function; Friends for functional notation, friend classes, this pointer; Accessing Member Data with this, using this for returning values.

Templates and Exception Handling: **6 Hrs**

Introduction, Templates, Class Templates, function templates, Member function templates, Template arguments, Exception Handling.

BOOKS :

1. Prata : C++ Primer Plus, 4/e Pearson Education
2. Lafore Robert : Object Oriented Programming in Turbo C++, Galgotia Publications
3. Lippman : C++ Primer, 3/e Pearson Education
4. E. Balaguruswamy : Object Oriented Programming with C++, Tata McGraw Hill Publications.
5. Strousstrup : The C++ Programming Language, Pearson Edition, 3rd Edition
6. Kamthane : Object Oriented Programming with ANSI and Turbo C++, Pearson Education
7. Bhave : Object Oriented Programming Using C++, Pearson Education

TITLE : Visual Programming
CODE : CA3215
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 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 if 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.

Introduction: **4 Hrs**

Windows Concepts, Objects and events, Define design and development process, Identify elements of IDE, Write, run, save, and print a project, Use online Help.

Introduce controls and their properties: **4 Hrs**

Text boxes, group boxes, check boxes, radio buttons, picture boxes and naming conventions, User friendly features: access keys, default and cancel buttons, tab sequence, Tool Tips, resetting focus, Changing properties at run-time, Concatenate strings.

Variables, constants and calculations: **4 Hrs**

Declaration of variables and constants considering data types and scope, Explicit data type conversions, Perform calculations using arithmetic operators and order of operations, Use of accumulators and counters, Use formatting functions to format output.

Decisions and conditions: **4 Hrs**

Use If statements to control the flow of logic, Use And and/or operators, Call event procedures, Input validation, Debug tools - set break points, stepping and observation of intermediate results.

Arrays: **2 Hrs**

Declare arrays and refer to elements using subscripts, Use For Each/Next statements, Structure Variables, Store data in multidimensional array.

Lists, Loops, and Printing: **2 Hrs**

Create and use list boxes and combo boxes, Use Do/Loops and For/Next statements, Send information to the printer.

Menus, procedures and functions: **6 Hrs**
Create menus and submenus for program control, Write reusable code in sub procedures and sub functions.

Toolbars and Status bar: **4 Hrs**
Creating toolbars, Adding images to toolbars, Writing code to work with toolbars, Creating and using a status bar, Adding panels to the status bar.

Other controls: **4 Hrs**
TabControl, MonthCalendar and Date/TimePicker, Common Dialog (Open/SaveFile), ProgressBar.

File handling and file controls in VB: **4 Hrs**

Database Connectivity: **12 Hrs**
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.

Introduction to Graphics and animation: **2 Hrs**

Multiple Document Interface (MDI) **4 Hrs**
Overview of MDI, Creating parent and child forms, Writing code for parent and child forms, Child window management, Creating applications.

Creating and using reports: **4 Hrs**
Printing with windows forms, Data reports, Using Crystal Reports.

BOOKS :

1. Deitel, Visual Basic 6 How to Program. Pearson Education
2. Neol Jerke, The Complete Reference Visual Basic 6, Tata McGraw Hill (1999).
3. Evangelas and Petroutsos, Mastering VB 6, 1st Edition, BPB Publications (2001).
4. V.K. Jain, Introduction to OOP and VB, Vikas Publishing House (2003)
5. Gottfried, Programming with Visual Basic, Schaum's Series - Tata McGraw Hill.
6. Reselman, Peasley and Pruchniak, Using Visual Basic 6, PHI (2000).

TITLE : Database Management Systems
CODE : CA3315
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.

Introduction: **6 Hrs**

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: **6 Hrs**

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.

Data Modeling Using Entity-Relationship Model: **6 Hrs**

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. Weak Entity Types and Drawing E- R Diagrams.

Index Structures for Files: **6 Hrs**

Single Level Ordered Indexes – Primary indexes, Clustering indexes and Secondary indexes. Multi-level indexes, Dynamic Multilevel indexes using B-trees (Introductory concepts). Hashing concepts.

Relational Data Model: **12 Hrs**

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.

SQL: **4 Hrs**

Queries, sub queries, correlated sub query, views, updation of a database through views, Update, Delete.

Transaction Processing Concepts and Concurrency Control Techniques: **6 Hrs**

Transaction and System concepts – Desirable properties of Transactions – Schedules and Recoverability. Lock-Based Protocols – Locks, Granting of Locks, and Two phase locking protocol and implementation of locking.

Data Base Administration: **6 Hrs**

Introduction to Database security issues, Discretionary Access Control Based on Granting/Revoking of Privileges and Multi-level security.

Distributed Databases: **8 Hrs**

Distributed database concepts, Data fragmentation, Replication, and Allocation Techniques for Distributed database design, Types of Distributed database systems. Introduction to Advanced Database concepts – Brief introduction to Data warehousing and Data mining.

BOOKS :

1. Elmasri & Navathe, Fundamentals of Database Systems (Fourth Edition), Pearson Education, 2003.
2. Karate, Introduction to Database Management System, Pearson Education 2004.
3. Abrahamsi. Silberschatag, Henry. F. Korth, S. Sudarshan, Database System Concepts, McGraw hill.
4. Jeffrey. D. Ullman, Principles of database system.
5. Oracle Press: ORACLE - Computer reference
6. C.J. Date, Introduction to database systems, Sixth Edition, Addison Wesley, 1995.
7. Raghu Ram Krishnan, Database Management Systems, Second Edition,. McGraw Hill, 2000.

TITLE : Software Engineering
CODE : CA3415
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. • Understand the UML notation. • 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 softwares by applying the software engineering principles.
- Implement developed software efficiently and effectively.

Introduction : **6 Hrs**

Software Products and Software process, Process models: Waterfall modal, Evolutionary Development, Bohemia's Spiral model, Overview of risk management, Process Visibility, Professional responsibility.

Computer based System Engineering : **4 Hrs**

Systems and their environment, System Procurement, System Engineering Process, System architecture modeling, Human Factors, System reliability Engineering.

Requirements and Specification: **8 Hrs**

The requirement Engineering Process, The Software requirement document, Validation of Evolution of requirements, Viewpoint – oriented & method based analysis, System contexts, Social 7 organizational factors, Data flow, Semantic, Object, models, Requirement definition, Requirement Specification, Non functional requirement.

Software prototyping: **4 Hrs**

Prototyping in software process, Prototyping techniques, User interfaces prototyping.

Software Design: **6 Hrs**

Design Process, Design Strategies, Design Quality, System Structuring, Control models, Modular decomposition, Domain Specific architecture.

Object Oriented and function oriented design: **6 Hrs**

Objects, object Classes and inheritance, Object identification, An object oriented design example, Concurrent Objects, Data flow design, Structural decomposition, Detailed Design, A Comparison of design Strategies.

User interface design: **4 Hrs**

Design Principles, User System interaction, Information Presentation, User Guidance, Interface Evaluation.

Software Reliability and reusability: **8 Hrs**

Software reliability metrics, Software reliability Specification, Statistical testing, Reliability Growth modeling, Fault avoidance & tolerance, Exception handling & defensive programming, Software development with reuse, Software development for reuse, Generator based reuse, Application System Portability.

Software Verification and Validation: **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.

Management Issues: **4 Hrs**

Project management, Quality management, Software cost estimation, Software maintenance.

BOOKS :

1. Ian Summerville, Software Engineering, 6th Edition, Pearson Publication Ltd. 2001
2. Roger Pressman, Software Engineering – A practitioner's approach (McGraw Hill).
3. Carlo Ghejgietal, Fundamentals of Software- Engineering, Pearson Education.
4. Panakaj Jalote, An Integrated Approach to Software Engineering – Narosa
5. Publishing house.

PRACTICALS

TITLE : C++ LAB
CODE : CA3P1
Hrs / Week : 2 Hrs
CREDITS : 1

PART A (Program related to C++ concepts)

1. Find the largest and second largest of four numbers.
2. Check whether a given date is valid.
3. Find the GCD and LCM of two numbers
4. Find N C R where ${}^{N}C_{R} = N ! / ((N-R)! * (R!))$
5. Generate the fibonacci series using arrays.
6. Interchange the values of two variables using reference variables.
7. Find the factorial of a number using function overloading.
8. Find whether a given number is prime using function overloading.
9. Calculate compound interest using default arguments where $CI = P * (1 + R / 100)^{T}$.
10. Check whether a given number is odd or even using inline functions.

PART B (Program related to OOPS concepts)

1. Write a program to create a student database for a class containing Name, Reg No, Class, Combination include the following
   a. Constructors b. Destructors c. Default constructors
   d. Input and Output functions and Input and Output for 5 people using different methods.
2. Create a class to hold information for a customer about his current-account and savings-account in a bank. Using friend functions find the total balance of both the account.
3. Write a program to overload the following operators.
   a. Binary operator ' +' to concatenate 2 strings and compare using '=='
   b. Relational operator ' < ' to find whether one date is less than other.
   c. Find the sum of 2 matrices using operator ' +'
   d. Find the next date of a given date using '++' operator
   e. Using '+'. '-', '*' to find the sum , difference and product of 2 complex numbers
5. Create a base class for a stack and implement push and pop operation. Include derived class to check for stack criteria's such as
   a. Stack is empty b. Stack is full

TITLE : VISUAL PROGRAMMING LAB
CODE : CA3P2
Hrs / Week : 2 Hrs
CREDITS : 1

1. Design an application to validate the user name and password and display message.
2. Design an application to change font style, size, color using a combo box.

3. Design a caculator.
4. Design an application to show usage of timer.
5. Design an application to calculate the area and circumference of a circle.
6. Design an application to find the sum of numbers.
7. Design an application to authenticate travel system using list and combo box.
8. Design an application to add and remove item from list box.
9. Design an application to demonstrate sequential file .
10. Design an application to find the area and perimeter of a square using subroutine.
11. Design an application to create front end and back to implement ADO connection.
12. Design an application to implement crystal report.
13. Design an application to implement arithmetic operation using subroutine.
14. Design an application to implement scroll bar to the change the font size of the label.
15. Design an application to implement Menu Editor.
16. Design an application to move image using timer.
17. Design an application to implement shapes/ images/ pictures.
18. Design an application to implement message box and input box.
19. Design an application to implement if then, if then else conditions.
20. Design an application to implement while, do while.

FOURTH SEMESTER

TITLE : Computer Graphics
CODE : CA4115
Hrs / Week : 4 Hrs
CREDITS : 4

Course Objective:

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 Outcome:

Students will understand 2D and 3D graphic techniques which will help them to proceed with their project development.

Graphics Systems: **6 Hrs**
Display Devices, Hard Copy Devices, Interactive Input Devices, Display Processors, and Graphic software.

Output Primitives: **10 Hrs**
Points & Lines, Line drawing algorithms, DDA & Bresenham's line algorithms, Circle generating algorithms, Ellipses. Attributes of output primitives, line type, line width, line color, area filling, Scan line algorithm.

Two Dimensional Transformations: **10 Hrs**
Basic transformations, translation, Scaling and Rotation. Matrix representations and homogeneous co-ordinates, composite transformation -translation, scaling and rotations. Raster methods for transformation.

Windowing and Clipping: **8 Hrs**
Windowing concepts clipping algorithms, line clipping, area clipping, Blanking window to view port transformations.

Interactive Input Methods: **6 Hrs**
Physical input devices : keyboards, touch panels, line pens, graphics tablets, joysticks, mouse, trackball, interactive picture construction techniques.

Three Dimensional Concepts: **6 Hrs**
Three–dimensional co–ordinate systems, three–dimensional display techniques, perspective and parallel projections, polygon surfaces, curved surfaces, octrees, three–dimensional transformations.

Hidden Surface and Hidden Line Removal: **8 Hrs**
Back–face removal, depth buffer method, scan line method.
Shading and Color Mode : **6 Hrs**

BOOKS:

1. Donald Hearn & M.Pauline Baker, Computer Graphics C Version, Pearson Educaiton/PHI
2. Computer Graphics – Steven Harrington, McGH
3. Dr.Venugopal K.R .Syeda Noor Fathima, H.S. Vemala, A programming Approach, programming with Fortran Pascal and C, TMH.
4. Principles of Interactive Computer Graphics – Newman and Sproull, McGraw Hill
5. Graphics Under C – Yeshwant Kanetkar, BPB Publications.
6. James D Foley, Adries Van Dam, Steven K Feiner, John F Hughes, Computer Graphics, Addison Wesley, 1997.
7. Cooley: The Essence of Computer Graphics Pearson Education

TITLE : Computer Networks – I
CODE : CA4215
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

Introduction: **8 Hrs**

Uses of Computer networks, network hardware, network software, reference models.

The physical layer: **10 Hrs**

The theoretical basis for data communication, guided transmission media, wireless transmission, communication satilites, the public switched telephone network, the mobile telephone system.

The data link layer: **10 Hrs**

Data link layer design issues, error detection and correction, elementary data link protocols, sliding window protocols, protocol verification.

The medium access control sub layer: **10 Hrs**

The channel allocation problem, multiple access protocols, Ethernet, wireless LAN, broadband wireless, Bluetooth.

The network layer: **10 Hrs**

Network layer design issues, routing algorithms- the optimality principle, shortest path routing, flooding, congestion control algorithms-general principles of congestion control, congestion prevention policies.

The transport layer: **6 Hrs**

The transport service, elements of transport protocols, a simple transport protocol.

The application layer: **6 Hrs**

DNS, electronic mail, the world wide web.

BOOKS:

1. Andrew S.Tannenbaum, "**Computer Networks**", Fourth Edition, Tata McGraw-Hill Publishing Company Limited NewDelhi
2. Behrouz A. Forouzan, "**Data Communications and Networking**", TATA Mc-
3. Grow-Hill publications, Second Edition, 2003.
4. William Stallings, "**Data & Computer Communications**", Pearson Education
5. Asia, Sixth Edition, 2001.
6. William A. Shay, "**Understanding Data Communication and Networks**", Vikas Publishing House, Second Edition, 2001.

TITLE : JAVA programming
CODE : CA4315
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.
- Produce sample use-cases, pseudocode, 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.

Introduction to JAVA: 12 Hrs

JAVA Evolution, Introduction to Programming Languages, The Evolution of Java, Object-Oriented Programming Concepts and Java, Differences between C++ 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. Constants, Variables, and Using Data Types, Operators and Expressions, Type conversion and Associativity, Mathematical Functions. Decision Making and Branching: 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.

Classes, Arrays, Strings and collection frame work: 12 Hrs

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, Finalizer methods, Abstract Methods and Classes, Visibility Control.

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

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.

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.

Managing Exceptions, Applet Programming: 12 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 Usr.

Managing Input / Output Files in JAVA: **8 Hrs**

Introduction, Concept of Streams, Stream Classes, Byte Stream Classes, Character Stream Classes, Using Streams, Other Useful I/O Classes, Using the File Class, Input/Output Exceptions, Creation of Files, Reading/Writing Characters, Reading/Writing Bytes, Handling Primitive Data Types, Concatenating and Buffering Files, Interactive Input and output, Other Stream Classes.

BOOKS:

1. Shishir Gundavaram, CGI Programming on the World Wide Web, O'Reilly and Associates, (1996).
2. E. Balaguruswamy, Programming with JAVA, A Primer, 2nd Edition., TMH (1999)
3. Thomas Boutel, CGI programming in C and Perl, Addison – Wesley, (1996).
5. Jefry Dwight et al, Using CGI, (Second Edition), Prentice Hall, India, (1997).
6. Darrel Ince & Adam Freeman, Programming the Internet with Java, Addison – Wesley, (1997).
7. KenArnold & James Gosling, The Java Programming Language, Addison – Wesley, (1998) 8. Patrick Naughton & Herbert Schildt, JAVA 2: The Complete Reference, 3rd Edition, TMH, (1999).

TITLE : Web Development
CODE : CAOE4415 (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 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.

Internet Basics: **10 Hrs**

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

Web Designing Technologies: **10 Hrs**

Introduction to HTML, ASP, JSP, Java scripts, VB scripts, HTML/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

Java Script: **5 Hrs**

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: **5 Hrs**

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

BOOKS

1. E. Balaguruswamy, Programming with JAVA, A Primer, 2nd Edition., TMH 2
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).
4. Internet 6-in-1 by Kraynak and Habraken, Prentice Hall of India Pvt. Ltd., New Delhi
5. Internet for Everyone by Alexis Leon and Mathews Leon; Vikas Publishing House Pvt. Ltd., New Delhi.

TITLE : Basic Programming Skills
CODE : CAOE4515 (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.

Introduction To Programming: **5 Hrs**

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: **5 Hrs**

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.

Control Structures: **10 Hrs**

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.

Arrays: **5 Hrs**

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

Functions: **5 Hrs**

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.

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 more than 10.
12. Generate one single thread.
    a) using Thread class
    b) using Runnable Interface.
13. To find factorial of list of number reading input as command line argument.
14. To find prime series reading N as command line argument.

15. To sort list of elements in ascending and descending order and show the exception handling.
16. To implement constructor overloading by passing different number of parameter of different types.
17. To create student report using applet, read the input using text boxes and display the o/p using buttons.
18. To calculate bonus for different departments using method overriding.

TITLE : Computer Graphics Lab
CODE : CA4P2
Hrs / Week : 2 Hrs
CREDITS : 1

1. Line using
   (a) DDA algorithm and
   (b) Bresenham's algorithm
2. Circle using
   (a) DDA algorithm and
   (b) Midpoint circle algorithm
3. Write a program showing
   (a) line styles and
   (b) area filling.
4. Translation and scaling of a rectangle.
5. Rotation of a triangle:
   (a) With respect to the origin.
   (b) With respect to a pivot point.
6. Reflection of a circle:
   (a) With respect to the origin.
   (b) with respect to X-axis
   (c) with respect to Y-axis
   (d) With respect to the line Y=X.
7. Shearing of a rectangle
   (a) With respect to x axis
   (b) With respect to y axis
8. Vertical and Horizontal histogram.
9. Pie-chart.
10. Implementation of Cohen-Sutherland line clipping algorithm.
11. Animation- Man walking with an umbrella.

FIFTH SEMESTER

TITLE : Computer Organization And Architecture
CODE : CA5115
Hrs / Week : 4 Hrs
CREDITS : 4

Course 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:

On successful completion of the course the students should have
- Understood Computer Architecture.
- Understood I/O, Registers and memory.
- Understood processor design, control unit design.
- Understood I/O interfacing.

Integrated circuits and digital functions: 10 Hrs
Digital integrated circuits, IC flip-flops and registers, decoders and multiplexers, binary counters, shift registers, random - access memories (RAM) read-only memories (ROM).

Data representation: 6 Hrs
Data types, fixed-point representation, floating-point representation, other binary codes, error detection codes.

Basic computer organization and design: 12 Hrs
Instruction codes, computer instruction, timing and control, execution and instruction, input-output and interrupt, design of computer.

Central processor organization: 12 Hrs
Processor bus organization, arithmetic logic unit (ALU), stack organization, instruction formats, addressing modes, data transfer and manipulation, program control, microprocessor organization.

Input-output organization: 10 Hrs
Peripheral devices, I/O interface, asynchronous data transfer, direct memory access (DMA), priority interrupt, input-output processor (IOP).

Memory organization: **10 Hrs**

Auxiliary memory, microcomputer, memory hierarchy, associative memory, virtual memory, cache memory, memory management hardware.

BOOKS:

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
CODE : CA5215
Hrs / Week : 4 Hrs
CREDITS : 4

Course Objective:

To know the different techniques and formats behind the multimedia technology.

Learning Outcome:

The student will be able to know the compression and decompression techniques along with the knowledge of various formats of files and their differences.

Multimedia Information Representation: **15 Hrs**

Introduction, Digitization Principles – Analog Signals, Encoder Design, Decoder Design.
Text: Unformatted Text, Formatted Text, Hyper Text.
Images: Graphics, Digitized Documents, Digitized Pictures.
Audio: PCM Speech, CD – Quality Audio, Synthesized Audio. Video – Broadcast Television, Digital Video, PC Video, Video Content.

Text Compression: **10 Hrs**

Compression Principles – Source Encoder and Destination Decoder, Lossless and Lossy Compression, Entropy Encoding, Source Encoding. Text Compression – Static and Dynamic Huffman Coding, Arithmetic Coding.

Continuous Media: **10 Hrs**

Pcm speech, cd-quality audio, synthesizers, musical instrument digital interface, broadcast television, digital video, pc video.

Image Compression: **8 Hrs**

Graphics Interchange Format (GIF), Tagged Image File Format (TIFF), Digitized Documents, JPEG.

Audio Compression: **9 Hrs**

Differential Pulse Coded Modulation (DPCM), Adaptive Differential PCM (ADPCM), Adaptive Predictive Coding and Linear Predictive Coding, MPEG Audio Coding.

Video Compression: **8 Hrs**

Principles, CD, DVD, H.261 Video Compression, MPEG 1, MPEG2 and MPEG 4.

BOOKS:

1. Halshall, Fred. "Multimedia Communications – Applications, Networks, Protocols and Standards". 2001. Pearson Education.
2. Chapman, Nigel and Chapman, Jenny. "Digital Multimedia". 2000. John Wily & Sons.
3. Steinmaetz, Ralf and Nahrstedt, Klara. Multimedia : "Comunications and Applications". 2003. Pearson Education.

TITLE : WEB TECHNOLOGIES
CODE : CA5315
Hrs / Week : 4 Hrs
CREDITS : 4

Course Objective:

To help the students to understand the concepts of HTML, CSS, Java script and PHP.

Learning Outcome:

The student will be able to completely develop a dynamic website with database backend.

Unit-1 Fundamentals of Web **12 Hrs**

Internet:

WWW, Web Browsers, and Web Servers, URLs, MIME, HTTP, Security. HTML and CSS HTML - XHTML – HTML 5.

Creating simple web page, basic text formatting, presentation elements, phrase elements, lists, font, grouping elements, basic links, internal document links, email link, Image, Audio and Video, image maps, image formats, Adding flash content and video, Tables – attributes, nested tables, Forms – Attributes, form controls, Frames – Frame set, nested frames, attributes. Introduction to HTML 5 - New tags of HTML 5 – embedding Media content, building input forms, painting on canvas.

Cascading Style Sheet:

Introduction, Levels of Style Sheet and specification formats, embedded style sheet, External Style Sheet, inline Style Sheet, Class and ID method, DIV and SPAN tags, Inheritance with CSS, Introduction to CSS 3, HTML 5 and CSS3.

Unit-2 **JavaScript** **12 Hrs**

Javascript:
JavaScript Implementation, JavaScript in HTML, language basics – variables, operators, statements, functions, data type conversions, reference types, Document Object Model - Browser Object Model - window object, location object, navigator object, screen object, history object, Events and Event handling, Button elements, Navigator object, validations with regular expressions. Introduction to dynamic documents, positioning elements, moving elements, elements visibility, changing colors and fonts, dynamic content, locating mouse cursor, reacting to a mouse click, dragging and dropping of elements, basic animation with image using JavaScript

Unit-3 **PHP** **12 Hrs**

PHP:
Introduction to Server Side Programming, Introduction to PHP, PHP and HTML, essentials of PHP, Why Use PHP, Installation of Web Server, WAMP Configurations, Writing simple PHP program, embedding with HTML, comments in PHP, variables, naming conventions, strings, string concatenation, string functions, float functions, Arrays, Array – key pair value, array functions, isset(), unset(), gettype(), settype(), control statements (if, switch), loops, user defined functions (with argument and return values), global variable, default value, GET & POST method, URL encoding, HTML

Encoding, Cookies, Sessions, Include statement, File – read and write from the file.

Unit-4 **MySql** **12 Hrs**

MySql:
Introduction to MySQL, CRUD - select statements, creating database/tables, inserting values, updating and deleting, PHP with MySQL, creating connection, selecting database, perform database (query), use returned data, close connections, file handling in PHP – reading and writing from and to FILE, Using MySQL from PHP (Building a Guestbook).

Unit-5 **Object Oriented Programming with PHP** **12 Hrs**

Object Oriented Programming with PHP:
Introduction to OOPS, creating classes, creating objects, setting access to properties and methods, constructors, destructors, overloading and overriding of methods. Accessing PHP and HTTP Data. Reading POST and GET variables, Form validation.

BOOKS:

1. Jon Duckett, Beginning HTML , XHTML, CSS, and JavaScript, Wiley Publishing, 2010.
2. Steve suehring, JavaScript Step by Step, Microsoft Press, 2nd Edition, PHI, 2012
3. Matt Doyle, Beginning PHP 5.3, Willey Publishing, 2010.
4. Faithe Wempen. HTML 5 Step by Step, Microsoft Press, PHI, 2012
5. David Sawyer McFarland, CSS – The Missing Manual, Pogue Press, O'Reilley Willey Publishing, 2nd Edition, 2009.

TITLE : Computer Network-II
CODE : CA5415
Hrs / Week : 4 Hrs
CREDITS : 4

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.

Local Area Network: 4 Hrs
Features, components of LAN, Benefits of network, LAN evaluation, planning and installing of a LAN.

TCP/IP: 4 Hrs
Origin, Layering, Internet Address, port numbers, DNS, client server model, RFC's services, API, Link layer, SLIP,MTU.

Internet Protocol: 14 Hrs
IP Header, routing, subnet addressing, masks, ARP Introduction, ARP Cache, ARP Packet format, Proxy ARP, RARP-RARP packet format, server design, ICMP message types, address mask request and reply unreachable code, Time stamp, request and reply, ping program, IP record route option, IP time stamp option, Trace execute program, Trace route program operation, IP source route option.

IP and Dynamic Routing: 14 Hrs
IP routing: principals, ICMP host and network unreachable errors, ICMP redirect errors, ICMP router discovery messages, Dynamic routing: RIP,OSPF,BGP,CIDR User Datagram Protocol, UDP header, checksum, IP fragmentation, ICMP unreachable error, Maximum UDP datagram size, ICMP source quench error, UDP server design , Introduction to broadcasting and multicasting, DNS: Introduction, basics, message format, query message, resource record, caching.

TCP: 12 Hrs
Services, header, connection establishment and termination, TCP state transition diagram, Reset segments, simultaneously open and close. TCP Interactive Data Flow: Interactive Input, delayed acknowledgements, Nagle's Algorithm. Bulk Data flow: Normal data flow, sliding window, window size, congestion . Time out and Re-transmission. Round trip time measurement.

Telnet and Remote Login: **12 Hrs**

Introduction, Rlogin protocol, TELNET protocol, FTP: Protocol, data representation, FTP commands, Replies, connection management. SMTP: Protocol, SMTP commands, MIME. NFS: Sun RPC, NFS protocol.

BOOKS:

1. Andrew S.Tannenbaum, "**Computer Networks**", Fourth Edition, Tata McGraw-Hill Publishing Company Limited NewDelhi
2. Behrouz A. Forouzan, "**Data Communications and Networking**", TATA Mc-
3. Grow-Hill publications, Second Edition, 2003.
4. William Stallings, "**Data & Computer Communications**", Pearson Education
5. Asia, Sixth Edition, 2001.
6. William A. Shay, "**Understanding Data Communication and Networks**", Vikas Publishing House, Second Edition, 2001

PRACTICALS

TITLE : WEB TECHNOLOGIES LAB
CODE : CA5P1
Hrs / Week : 2 Hrs
CREDITS : 1

Guidelines:

- The output of the programs should be neatly formatted.
- The source code should be indented.
- The programs need to be interactive.
- Data validations can be done wherever applicable.
- Include comments to improve the readability of the program .
- Use meaningful variable names.
- Program should be prepared by their own.
- Follow the ethics of Programming, Web Design and Development.

1. Create a Web page by making use of the following tags : Headers, Linking and Images.
2. Create a Web page that will have the following: Frames, Unordered Lists, Nested and ordered Lists
3. Create a Web page Layout with Tables and all its attributes
4. Create a Web page that will have Application form (Forms) , make use of Image Maps and <meta> Tags
5. Create an External Style Sheet that defines the style for the following tag : H1, H2, Body , P, Li .
6. Create an Internal Style Sheet that defines a style for Positioning elements & setting the background (color / image)
7. Create a Style Sheets that defines the style with class method , Id method , make use of DIV and Span TAG
8. Create a style Sheet that demonstrate Box Model

9. Write a JavaScript program to Demonstrate the use of Variable , message box , and loops
10. Write a JavaScript Program to demonstrate Functions (predefined / user defined)
11. Write a JavaScript program to demonstrate Event Handling
12. Object Creation and modification in JavaScript
13. Write a PHP program to demonstrate GET and POST method of passing the data between pages
14. Write a PHP program to demonstrate Array , Key-pair values
15. Write a PHP program to read and write the Data from the Database
16. Create a PHP page that uses Session and cookies.
17. File Handling in PHP
18. Implementing the OOPs concept in PHP

TITLE : MINI PROJECT
CODE : CA5P2
Hrs / Week : 2 Hrs
CREDITS : 1

Mini Project Lab Guidelines

Students will be required to pursue a project work allotted to them. This work generally involves solving some practical problem, developing computer programs using the knowledge acquired in the theory and laboratory courses. They will have to submit a report of the work done by them.

SIXTH SEMESTER

TITLE : E-commerce
CODE : CA6115
Hrs / Week : 4 Hrs
CREDITS : 4

Course Objectives:

- To understand the nature and current trends of e-Commerce;
- Recognize the business impact and potential of e-Commerce;
- Explain the technologies required to make e-Commerce viable
- Discuss the current drivers and inhibitors facing the business world in adopting and using E-Commerce; and explain the economic consequences of e-Commerce;
- Discuss the trends in e-Commerce and the use of the Internet.
- Understand the importance of advertising and marketing in the field of e-commerce.

Learning Outcome:

By the end of this course, students should be able to discuss e-commerce from an enterprise point of view, evaluate key aspects of B2B e-commerce, discuss emerging e-commerce topics, understand business applications of electronic data interchange, understand the concepts of different electronic payment system, understand the concepts of information based marketing and the functioning of an e-market.

Introduction: 15 Hrs

Electronic Commerce Environment and opportunities: Background. The electronic Commerce environment, Electronic marketplace technology. Modes of electronic commerce: Overview, Electronic Data Interchange (EDI), Migration to open EDI, e-commerce with Internet/WWW, Commerce Net advocacy, Web Commerce going Forward. Approaches to safe e-commerce: Overview, Secure Transport Protocols, Secure Transactions, Secure Electronic Payment Protocol, Secure Electronic Transaction, Certificates, for Authentication, Security on Web Servers and Enterprise Networks.

Electronic Payment system: 10 Hrs

Electronic Payment systems: Types, Digital Token-Based Electronic Payment Systems, Smart Cards and Electronic Payment Systems, Credit card-based Electronic Payment Systems, Risk and Electronic Payment Systems, Designing Electronic Payment Systems.

Securing Electronic Transactions : 9 Hrs

Introduction, Business Requirements, Concepts, Payment Processing e-mail and Secure e-mail Technologies: Introduction, The means of Distribution. A Model for Message Handling. Working of e-mail. MIME, S/MIME: and MOSS, Comparisons of security methods.

Applications – 1: **12 Hrs**

Consumer-Oriented e-Commerce: Applications, Mercantile Process Models, Mercantile Models from the Consumer's Perspective and from the Merchant's Perspective. Interorganizational Commerce and EDI: EDI, Applications of EDI in business, Legal, Security and Privacy Issues; EDI standards and initiatives. EDI Software Implementation, EDI Envelope for Message Transport, Value-added Networks.

Applications – II: **14 Hrs**

Advertising and Marketing: The New Age of Information – Based Marketing, Advertising on the Internet, adaptation of new product: introduction process to the internet Marketing Research. Consumer Search and Resource Discovery: Search and Resource Discovery Paradigms. Information Search and Retrieval, e-Commerce Catalogs or Directories.

BOOKS:

1. Deitel, Internet and World Wide Web How to Program, Pearson Education, Asia.
2. Daniel Minoli, Emma Minoli, Web Commerce Technology Handbook, Tata McGraw Hill, (1998) (Chapter 1,2,3,6,7,11)
3. Ravi Kalakota, Andrew B. Whinston, Frontiers of Electronic Commerce, Addison-Wesley, (1996), (Chpater 1,7,8,9,10,11,13,14)
4. Daniel Lynch and Leslie Lundquist, Digital Money: The New Era of Internet Commerce, John Wiley, (1996)
5. Laudon, E-Commerce, Pearson Education, Asia

TITLE : Object Oriented System Design
CODE : CA6215
Hrs / Week : 4 Hrs
CREDITS : 4

Course Objective:

The course 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 Outcome:

After successfully completing this course you will be able to explain:

1. The object-oriented systems life cycle
2. Object-oriented methodologies, especially the Unified Modeling Language
3. Object-oriented analysis
4. Object-oriented design of classes and methods

Introduction: **15 Hrs**

An overview of Object oriented Systems Development: Object Orientation, Object Basics: An Object-Oriented Philosophy, Objects, grouping objects in classes, Attributes; Object Behavior

and methods, Encapsulation and Information Hiding, Class Hierarchy, Polymorphism, Object Relationships, and Associations, Aggregations and Object Containment. Object-Oriented Systems Development Life Cycle: Introduction. The software Development Process, Building High Quality software.

Methodology Modeling And Uml: 15 Hrs

Object Oriented Methodology: Introduction, Rumbaugh et al's Object Modeling Technique, The Booch Methodology, The Jacboson et al methodologies, Patterns. Unified Modeling Language: Introduction, Static and Dynamic Models, UML Diagrams, UML class Diagram, Use Case Diagram, UML Dynamic Modeling, Model Management: Packages And Model Organization, UML Meta- Model

Object Oriented Analysis: 15 Hrs

Identifying Use Cases: Introduction, Business Object Analysis: Understanding the Business Layer, Use-Case Driven Object - Oriented Analysis: the Unified Approach, Business Process Modeling, Use – Case Model, Developing Effective Documentation. Object Analysis: Classification introduction, Classification Theory, Approaches For Identifying Classes, Noun Phrase Approach, Classes, Responsibilities and Collaborators process, Naming Classes, Identifying Object Relationships, Attributes, and Methods :associations, Super-Sub Class Relationships, A-part-of Relationships-Aggregations, Class Responsibility: Defining Attributes by analyzing Use Case and other UML Diagrams, Object Responsibility: Methods and messages,

Object Oriented Design: 15 Hrs

The Object Oriented Design Process, Object Oriented Design Axioms, Corollaries, Design patterns, Designing Classes: The Process, Class Visibility; Designing Well-Defined Public, Private and Protected protocols, Designing Classes: Refining Attributes, Designing Methods and Protocols, Packages and managing Classes, Access Layer: Object Store and Persistence: Database Management systems, Object Oriented Database Management Systems, Object relational systems, View Layer designing Interface Objects: View Layer Classes, Macro-Level Process, Micro level Process.

BOOKS:

1. Ali Bahrami : Object Oriented Systems Development, McGraw hill,1999.
2. Booch: Object Oriented Analysis and Design Pearson Education
3. Criag Larman: Applying UML and Patterns, an Introduction to Object –Oriented Analysis and Design. Pearson Education 1998
4. Rebecca Wirfs-Brock et al: Designing Object-Oriented software, Prentice-Hall India 1990
5. Grady Booch: Unified modeling Language User Guide, Pearson Education,
6. Gamma: Design patterns: Elements of Reuasable Object Oriented Software , Pearson Education
7. Shalloway: Design Patterns Explained Pearson Education

TITLE : .NET Technologies
CODE : CA6315
Hrs / Week : 4 Hrs
CREDITS : 4

Course Objective:

In this course, you will use Visual Studio .NET and ASP .NET to build powerful Web sites. You will also use Access and Microsoft Sql Server 2005 for database connection. You will also be introduced to XML files to create XML. This will allow you to build sites that access databases and process data using dynamic, server-side programming. Thanks to these newly learned skills, your sites will be accessible to users of any of the popular browsers.

Learning Outcome:

- Upon successful completion of this course, students will be able to: create a Microsoft ASP.NET Web Application Form, add code to a Microsoft ASP.NET Web form, validate user input, create user controls, and database connection.

Introduction to .NET: 4 Hrs

.NET Definition, Advantages of .NET, .NET Architecture – Common Language,Runtime, MSIL, Support of different Languages. Language Interoperability, .NET Framework Classes. Advantages of Managed Code – Strong Data Type Check, Garbage Collection, Security, Performance Improvement.

C# Basics: 8 Hrs

Features of C# – Data types, Flow Control – the Main method, Program Structure, Methods, Arrays, Namespaces.

Object Oriented C#: 10 Hrs

Classes and Inheritance, Method Overloading, Method Overriding, Calling Base Versions of Methods. Abstract Classes and Methods, Sealed Classes and Methods. Access Modifiers. Properties – Read Only, Write Only Properties. Function – Parameter Passing Mechanisms. Interfaces, Dispose methods. Operator Overloading, Indexers.

Advanced C# Topics: 12 Hrs

Errors and Exception Handling, Exception Classes, User Defined Exceptions. The STD namespace objects, Array Lists, Collections, Dictionaries. Multi Threading –
Synchronization. Delegates – Definition, Delegates in Inheritance. Event handler, Reflection.

Programming in the .NET Environment: 6 Hrs

Introduction to Visual Studio .NET – ASP .NET. Difference between ASP and ASP.NET. Creating a Web application using ASP.NET. Components of an ASP.NET User Control, Custom Control, Deploying ASP .NET applications. Master Pages, Themes.

Assemblies: **6 Hrs**

Features of Assemblies, Application Domains, Assembly Structure, Assembly manifests, Assemblies and Components.

Data Access: **14 Hrs**

ADO.NET overview. Various data access objects – Connection, Command and DataSet Objects. Binding data to ASP .NET server controls. Accessing data from a database using ADO.NET. Reading from and Writing to an XML document, Using XML DOM objects for data access from XML Documents. Binding data from an XML document to Web form controls. Converting data from Database to XML Data. Xml & Web Services.

BOOKS:

1. Simon Robinson, Christian Nagel, Karli Watson, Jay Glynn, Morgan Skinner and Bill Evjen, Professional C#, Wiley – dreamtech India Pvt. Ltd., 3rd Edition, 2004.
2. .NET(Core Reference) Microsoft® Visual C#® 2005: The Language by Donis Marshall
3. Complete-reference-to-professional-soa-with-visual-studio-2005-dot-net-3-0
4. Kothari Nikhil and Datye Vandana, Developing ASP .NET Server Controls and Components, Tata McGraw Hill, 2003.
5. Esposito Dino, Applied XML Programming for Microsoft .NET, Tata McGraw Hill, 2003

TITLE : Mobile Applications
CODE : CA6415
Hrs / Week : 4 Hrs
CREDITS : 4

Course Objective:

This course focuses on developing applications for modern Smartphone operating systems.

- To provide a practical approach to Android mobile application development.
- To make students understand, how to develop and deploy an application to the app market.

Learning Outcome:

Upon completion of this course, Students will be able to:

- Explain the differences between different mobile technologies.
- Building Android applications.
- Secure, tune, package and deploy Applications.

Overview: **15 Hrs**

A little background about mobile technologies, Different mobile technologies – Android, Windows, IOS, Black Berry, series 40, Bada, Benefits and drawbacks of Smartphone programming, Overview of Android, How it all got started, Why Android different and

important, Android Stack overview, Linux kernel, native libraries, App framework, Apps, SDK overview, platforms, tools, versions. Creating and setting up custom Android emulator

Android an Overview: **15 Hrs**

Install the android SDK, Install base tools, install SDKs and Addons, Install apache Ant, Emulator, and Device. Get know Eclipse, Build , install and Run the Application in your Emulator or Device, Project Structure.

Designing User interface: **15 Hrs**

Designing by declaration, creating the opening screen, using alternate resources, implementing an about box, applying a theme, adding a menu, adding settings, debugging with log messages, debugging with debugger

Exploring 2D Graphics and Multimedia: **8 Hrs**

Learning the basics, adding Graphics to existing apps, handling input, learn to change the final improvements, Playing audio, Playing Video, Adding sound to existing app,

Storing local Data: **7 Hrs**

Reading/writing local data, Accessing the Internal File system, Accessing SD card.

BOOKS:

1. Grant Allen, Beginning Android 4, Apress, 2012.
2. WeiMeng Lee, Beginning android 4 application Development, John Wiley & sons, Inc, 2012.
3. Ed Burnette, Hello, Android: Introducing Google's Mobile Development Platform,Pragmatic. Bookshelf (2009), ISBN13:9781934356173.
4. Jerome (J.F) DiMarzio , Android A programmer's Guide, TataMcgraw Hill ,2010, ISBN: 9780071070591.

PRACTICALS

TITLE : .NET programming
CODE : CA6P1
Hrs / Week : 2 Hrs
CREDITS : 1

C# Programs

1. To Check whether a number is Palindrome or not.
2. To demonstrate Command line arguments Processing.
3. To find the roots of Quadratic Equation.
4. To demonstrate Operator overloading.
5. To multiply to matrices using Rectangular arrays.
6. To reverse a given string using C#.
7. Use Try, Catch and Finally blocks to demonstrate error handling.
8. Demonstrate Use of Virtual and override key words in C# with a simple program

9. To build a class which implements an interface which is already existing.

ASP.NET Programs

1. Design an application to validate the user name and password and display message.
2. Design an application to change font style, size, color using a combo box.
3. Design a calculator.
4. Design an application to show usage of timer.
5. Design an application to find the sum of numbers.
6. Design an application to add and remove item from list box.
7. Design an application to create front end and back to implement ADO connection.
8. Design an application to implement crystal report.
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.
11. Design an application to implement Menu Editor.
12. Design an application to move image using timer.

TITLE : MAJOR PROJECT
CODE : CA6P2
Hrs / Week : 2 Hrs
CREDITS : 1

Major Project Lab

Students will be required to pursue a project work for an organization of their choice with the permission of the HOD. This work generally involves collecting data, solving and implementing a problem for the organization, developing computer programs using the knowledge acquired in the theory and laboratory courses. They will have to submit a report of the work done by them. Finally a demonstration of the work with the help of a presentation has to be done.