SACRED HEART COLLEGE (AUTONOMOUS), THEVARA

KOCHI, KERALA, 682013

CURRICULUMAND SYLLABI

CHOICE BASED COURSE CREDIT AND SEMESTER SYSTEM

(CBCSS)

UNDERGRADUATE PROGRAMME IN COMPUTER APPLICATION

(INTRODUCED FROM 2015 ADMISSION ONWARDS)

BOARD OF STUDIES IN COMPUTER SCIENCE

Sacred Heart College (Autonomous), Thevara, Kochi, Kerala

CONTENTS

1.1 RATIONALE OF PROGRAMME

The radical changes in technologies, both hardware as well as software, and their ever increasing adaptation to newer areas of application, demand frequent updating of the academic curriculum so that the students can rise to the expectation of the Industry. The syllabus revision committee has considered all these factors thoroughly before venturing into the revision exercise.

BSc Computer Application graduates can start up their career in either Government sector or private sector since there are a lot of employment opportunities in both these sectors. This degree helps the students to get a sound knowledge on computer applications with the help of which they can have a perfect launch pad for challenging career in the field of information technology.

BSc Computer Application, triple main UG, self-financing vocational programme was designed and approved by the UGC. This contains the subjects Computer Science, Statistics and Mathematics. After the successful completion of this programme, the students can join MCA, MSc Computer Science, MSc IT, MSc Mathematics andMSc Statistics.

1.1.1 JOB OPPORTUNITY:

We know that India is an IT power thus it is obvious that the course of BSC CA will provide jobs in the IT sector. The companies such as TCS, Wipro, Infosys are hiring Students every year. It is of out sounding news to know that even the Students are getting placements in banking firms. If one wishes to become a Software Developer or web-designer or tester or wants to take up a career in Systems Management, then this course is the best course to get an early start for their careers. Job opportunities can be found in both large and small software development organizations as well as hardware companies.

1.2 OBJECTIVES:

The programme

- Provides students with skills in problem statement, analysis, solution, implementation and testing.
- Prepares the students the ability to adapt to various computer hardware and software platforms.
- Equips students with relatively advanced analytical skills in Computer Science, statistics and Mathematics.
- Provides students with skills for careers in the public and private sectors.

- Prepares students for post-graduate study in any one of the three fields.
- Provides knowledge and insight to behave ethically in professional practice to support the larger community.

1.2.1 General Learning Outcomes:

Students will

- Understand the nature of scientific enquiry and research.
- Be able to analyse a problem, construct alternate approaches to its solution and evaluate the merits and demerits of each.
- Be able to communicate effectively, both orally and in writing.
- Recognise the need for life-long learning and development.

1.2.2 Specific Learning Outcomes:

Students will

- Understand the nature of the software development process, including the need to provide appropriate documentation.
- Be able to program fluently in one or two programming languages and understand the major programming paradigms and be able to learn a new programming language in a short time.
- Understand standard techniques for solving a problem on a computer, including programming techniques and techniques for the representation of information.
- Understand the basic theory of computer architectures, including computer hardware and networking and understand the importance and the nature of operating systems and compilers.
- Possess the skills to apply statistical procedures and modeling approaches to a wide variety of real-life problems and be able to develop an effective sampling plan.
- Have background in higher level mathematics that allows them to teach competently and confidently.

1.3 DEFINITION:

1.3.1 **'Programme'** means a three year programme of study and examinations spread over six semesters, according to the regulations of the respective programme, the successful completion of which would lead to the award of a degree.

1.3.2 **'Semester'** means a term consisting of a minimum of 450 contact hours distributedover90 working days, inclusive of examination days, within 18 five-day academic weeks.

1.3.3 **'Academic Week'** is a unit of five working days in which distribution of work is organized from day-one today-five, with five contact hours of one hour duration on each day. A sequence of 18 such academic weeks constitutes a semester.

1.3.4 **'Common Course I'** means a course that comes under the category of courses for English a selection of this is compulsory for all students undergoing undergraduate programmes.

1.3.5 **'Core course'** means a course in the subject of specialization within a degree programme.

1.3.6 **'Complementary Course'** means a course which would enrich the study of core courses.

1.3.7 **'Open course'** means a course outside the field of his/her specialization, which can be opted by a student.

1.3.8 **'Additional core course'** means a compulsory course for all under graduate students (as perthe UGC directive) to enrich their general awareness.

1.3.9 **'Additional Course'** is a course registered by a student over and above the minimumrequired courses.
1.3.10 **'Credit'**is the numerical value assigned to a course according to the relative importance of the content of the syllabus of the programme.

1.3.11 **'Additional credit'** is the numerical value assigned to Club activities, Social service, Internship etc. which is not added with the total academic credits of the students.

1.3.12 **'Internship'** is job training for professional careers.

1.3.13 **'College Coordinator'** is a teacher nominated by the College Principal to co-ordinate the continuous evaluation undertaken by various departments within the college.

1.3.14 **'Department'** means any teaching department in a college.

1.3.15 **'Parent Department'** means the department which offers core courses within a degree programme.

1.3.16 **'Department Council'** means the body of all teachers of a department in a college.

1.3.17 **'Department Coordinator'** is a teacher nominated by a Department Council to co-ordinate the continuous evaluation undertaken in that department.

1.3.18 **'Faculty Advisor'** means a teacher from the parent department nominated by the Department Council, who will advise the student in the choice of his/her courses and other academic matters.

1.3.19 **'Grace Marks'** shall be awarded to candidates as per the University Orders issued from time to time.

1.3.20 **'Grade'** means a letter symbol (e.g., A, B, C, etc.), which indicates the broad level of performance of a student in a course/ semester/programme.

1.3.21 **'Grade Point' (GP)** is the numerical indicator of the percentage of marks awarded to a student in a course. Words and expressions used and not defined in this regulation shall have the same meaning assigned to them in the Act and Statutes.

1.4 DURATION:

- The duration of U.G. programmes shall be 6 semesters.
- The duration of odd semesters shall be from June to October and that of even semesters from November to March.
- A student may be permitted to complete the Programme, on valid reasons, within a period of 12 continuous semesters from the date of commencement of the first semester of the programme.

1.5 REGISTRATION:

The strength of students for each course shall remain as per existing regulations, except in case of open courses for which there shall be a minimum of 15 and maximum of 75 students per batch, subject to a marginal increase of 10. For non-core compulsory courses the student strength shall be decided by the Academic Council of the College from time to time.

Those students who possess the required minimum attendance and progress during a semester and could not register for the semester examination are permitted to apply for Notional Registration to the examinations concerned enabling them to get promoted to the next semester.

1.6 SCHEME AND COURSES:

The U.G. programmes shall include (a) Common courses I & II, (b) Core courses, (c) Complementary Courses, (d) Open Course (e) Additional core course. (f) Study tour (g) Internship

1.6.1Additional Credit Components

- Talent & career club activity (optional)
- Social service (mandatory)
- Internship for Commerce, Communication and Computer Application (mandatory).
- Internship (desirable for other programmes)

1.7 PROGRAMME STRUCTURE FOR MODULE 3

A	Programme Duration	6 Semesters
B	Minimum credits required from common course	08
C	Minimum credits required from three Core courses including Project	109
D	Minimum credits required from Open course	03
E	Additional core course (Environmental studies)	04
	Total Credits required for successful completion of the programme	124
F	Club activity (desirable)	01
G	Social service (mandatory)	01
H	Internship (mandatory)	02
I	Virtual Lab (desirable)	01
J	Minimum attendance required	75%

1.8 EXAMINATIONS:

The evaluation of each course shall contain two parts:

- CONTINUOUS INTERNAL ASSESSMENT (CIA)

- END-SEMESTER EXAMINATION (ESE)

The internal to external assessment ratio shall be 1:3, for both courses with or without practical. There shall be a maximum of 75 marks for external evaluation and maximum of 25 marks for internal evaluation.

1.9 COMPUTATION OF GRADE AND GRADE POINTS:

For all courses (theory & practical), grades are given on a 07-point scale based on the total percentage of marks. (CIA+ESE) as given below

Percentage of Marks	Grade	Grade Point
90 and above	A+ - Outstanding	10
80-89	A - Excellent	9
70-79	B - Very Good	8
60-69	C - Good	7
50-59	D - Satisfactory	6
40-49	E - Adequate	5
Below 40	F - Failure	0

Note: Decimal are to be rounded to the next whole number

1.9.1 Computation of SGPA

The SGPA is the ratio of sum of the product of the number of credits with the grade points scored by a student in all the courses and the sum of the number of credits of all the courses undergone by a student in a semester.

$$\textbf{SGPA (Si)} = \sum(Ci \times Gi) / \sum Ci$$

Where Ciis the number of credits of the ith course and Gi is the grade point scored by the student in the ith course.

1.9.2 Computation of CGPA

The CGPA is also calculated in the same manner taking into account all the courses undergone by a student over all the semesters of a programme, i.e.

$$\mathbf{CGPA = \sum(Ci \times Si) / \sum Ci}$$

Where Si is the SGPA of the ith semester and Ci is the total number of credits in that semester.

Note: The SGPA and CGPA shall be rounded off to 2 decimal points and reported in the transcripts.

Illustration of Computation of SGPA and CGPA and Format for Transcripts

Computation of SGPA and CGPA

Illustrationfor SGPA

Course	Credit	Grade letter	Grade point	Credit Point (Credit x Grade
Course 1	3	B	8	3 X 8 = 24
Course 2	4	C	7	4 X 7 = 28
Course 3	3	D	6	3 X 6 = 18
Course 4	3	A+	10	3 X 10 = 30
Course 5	3	E	5	3 X 5 = 15
Course 6	4	D	6	4 X 6 = 24
	20			139

Thus, SGPA =139/20 =6.95

Illustrationfor CGPA

Semester 1	Semester 2	Semester 3	Semester 4
Credit : 20 SGPA:6.9	Credit : 22 SGPA:7.8	Credit : 25 SGPA: 5.6	Credit : 26 SGPA:6.0
Semester 5	Semester 6		
Credit : 26 SGPA:6.3	Credit : 25 SGPA: 8.0		

Thus, $$\mathbf{CGPA = \frac{20 \times 6.9 + 22 \times 7.8 + 25 \times 5.6 + 26 \times 6.0 + 26 \times 6.3 + 25 \times 8.0}{144} = 6.73}$$

Grades for the different semesters and overall programme are given based

On the corresponding SGPA/ CGPA as shown below:

SGPA/CGPA	Grade
Above 9	A+ - Outstanding
Above 8, but below or equal to 9	A - Excellent
Above 7, but below or equal to 8	B -Very Good
Above 6, but below or equal to 7	C – Good
Above 5, but below or equal to 6	D – Satisfactory
Above 4, but below or equal to 5	E – Adequate
4 or below	F – Failure

Note: A separate minimum of 30% marks each for internal and external (for both theory and practical) and aggregate minimum of 40% are required for a pass for a course.

For a pass in a programme, a separate minimum of Grade E is required for all the individual courses. If a candidate secures F Grade for any one of the courses offered in a Semester/Programme only F grade will be awarded for that Semester/Programme until he/she improves this to E grade or above within the permitted period. Candidate secure E grade and above will be eligible for higher studies.

1.10. DETAILED DISTRIBUTION OF COURSES

Choice-based Credit and Semester System: BSc Computer Application Programme – Model III

SEM	COURSE CODE	TITLE OF THE COURSE	NO. OF HOURS / WEEK	NO. OF CREDITS	WEIGHTAGE	
					INTERNAL	EXTERNAL
1	15U1CCENG1	Communication Skills	5	4	25	75
	15U1CRCAP01	Fundamentals of Digital System	4	3	25	75
	15U1CRCAP02	Programming in ‘C’	4	3	25	75
	15U1PRCAP1	Programming in ‘C’ (Lab)	4	2	25	75
	15U1CRCMT1	Foundation of Mathematics	4	3	25	75
	15U1CRCST1	Descriptive Statistics	4	3	25	75
2	15U2CCENG2	Critical Thinking, Academic Writing & Presentation	5	4	25	75
	15U2CRCAP03	Micro Processors & Computer Organization	4	3	25	75
	15U2CRCAP04	Data Structures Using ‘C’	4	3	25	75
	15U2PRCAP2	Data Structures Using C (Lab)	4	3	25	75
	15U2CRCMT2	Analytic Geometry, Trigonometry & Matrices	4	3	25	75
	15U2CRCST2	Probability and Statistics	4	3	25	75
	15U2ARENV1	Environmental Science	5	4	25	75
3	15U3CRCAP05	Data Communication and Computer Networks	4	3	25	75
	15U3CRCAP06	Object Oriented	4	3	25	75

		Programming In C++				
	15U3CRCAP07	System Analysis and Design	4	3	25	75
	15U3PRCAP3	Object Oriented Programming In C++ (Lab)	3	2	25	75
	15U3CRCMT3	Calculus	5	4	25	75
	15U3CRCST3	Probability Distributions	5	4	25	75

SEM	**COURSE CODE**	**TITLE OF THE COURSE**	**NO. OF HOURS / WEEK**	**NO. OF CREDITS**	**WEIGHTAGE**	
					INTERNAL	**EXTERNAL**
4	15U4CRCAP08	Advanced Web Technology Using SQL Server	4	4	25	75
	15U4CRCAP09	Database Management Systems	4	3	25	75
	15U4PRCAP4	Advanced Web Technology Using SQL Server (Lab)	2	2	25	75
	15U4CRCMT4	Vector Calculus, Theory of Equations & Numerical Methods	5	4	25	75
	15U4CRCST4	Statistical Inference	5	4	25	75
	15U4CRCST5	Sample Survey Analysis and Design of Experiments	5	4	25	75
5	15U5CRCAP10	Java Programming and Dynamic Webpage Designing	4	4	25	75
	15U5CRCAP11	Internet, Web Designing And Cyber Laws (Open Course)	4	3	25	75
	15U5PRCAP5	Java Programming and Dynamic Webpage Designing (Lab)	2	3	25	75
	15U5CRCMT5	Mathematical Analysis	5	4	25	75
	15U5CRCMT6	Differential Equations	5	5	25	75

	15U5CRCST6	Statistical Quality Control and Operations Research	5	4	25	75
6	15U6CRCAP12	Operating Systems	5	4	25	75
	15U6CRCAP13	Cyber Security	5	4	25	75
	15U6PRCAP6	Project	5	4	25	75
	15U6CRCMT7	Real Analysis	5	4	25	75
	15U6CRCST7	Computer Aided Data Analysis using Excel and R	5	4	25	75

* One course to be selected from the list of Open Courses.

Total credits for core and complementary	**-**	**109**
Additional core course		**04**
Open	**-**	**03**

		116
Total credits for English	**-**	**08**

	Total	**124**

1.11. MARKS DISTRIBUTION FOR EXTERNAL EXAMINATION AND INTERNAL EVALUATION

Marks distribution for external and internal assessments and the components for internal evaluation with their marks are shown below: Components of the internal evaluation and their marks are as below.

For all courses without practical

- Marks of external Examination : 75
- Marks of internal evaluation : 25

All the three components of the internal assessment are mandatory. For common course English in I Semester, internal oral examination shall be conducted instead of test paper.

Components of Internal Evaluation	Marks
Attendance	5
Assignment (Written assignments, preparation of models, charts, posters etc., field survey, field work)	5
Seminar/Viva	5
Test papers-2	10
Total	25

Project Evaluation: (Max. marks100)

Components of Project-Evaluation	Marks
Dissertation	50
Internal	25
Presentation	25
Total	100

Attendance Evaluation

For all courses without practical

% of attendance	Marks
90 and above	5
85 – 89	4
80-84	3
76-79	2
75	1

(Decimals are to be rounded to the next higher whole number)

1.12. CONDONATION OF SHORTAGE OF ATTENDANCE

Candidate can seek condonation of shortage of attendance only once in a 2 year course and twice in other courses of longer duration. Following are the rules regarding attendance requirement:-

Every candidate is to secure 75% attendance of the total duration of the course.

A candidate having a shortage of 10% can apply for condonation of shortage in prescribed form on genuine grounds. Condonation of shortage of attendance if any should be obtained at least 7 days before the commencement of the concerned semester examination.

It shall be the discretion of the Principal to consider such applications and condone the shortage on the merit of each case in consultation with the concerned course teacher and HoD. Unless the shortage of attendance is condoned, a candidate is not eligible to appear for the examination.

GRIEVANCE REDRESSAL MECHANISM

In order to address the grievance of students regarding Continuous internal assessment (CIA) a three-level Grievance Redressal mechanism is envisaged. A student can approach the upper level only if grievance is not addressed at the lower level.

Level 1:

At the level of the concerned course teacher

Level 2:

At the level of a department committee consisting of the Head of the Department, a coordinator of internal assessment for each programme nominated by the HoD and the course teacher concerned.

Level 3:

A committee with the Principal as Chairman, Dean of concerned Faculty, HOD of concerned department and one member of the Academic council nominated by the principal every year as members.

SEMSTER I

FUNDAMENTALS OF DIGITAL SYSTEMS

Hours/Week: 04 **15U1CRCAP01** **Hours/Semester: 60**

Course Description:

This subject deals with the basic concepts of functioning of a computer. The subject starts with an introduction to number systems and its applications in computers. The subject exposes the students to basic concepts of flip-flops, logic gates and design of different types of flip flops and counters. The discussion about working of devices like encoders and decoders, multiplexers and de multiplexers are dealt here. The design of half adders and full adders are also included as part of this subject. Classification of memory, registers and flags are also dealt with.

Objectives:

- Familiarize the working of functional units of computer
- Learn the Concepts of Boolean logic and digital logic circuit
- Analyze and design combinational and sequential digital systems. Use different techniques, among them a hardware description language and a functional programming language, to design digital systems.

Learning Outcomes:

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

- knowledge about number system
- apply knowledge of the basic digital registers
- Explain the basic Structure and operation of a digital computer
- apply knowledge of Combinational and Sequential circuit

UNIT I **(12 Hrs)**

Fundamentals: Introduction to Computer, History and generation, functional units, Hardware: CPU, Primary and Secondary storage, I/O devices, Software: System and Application Programming Languages: Machine Language, Assembly Language, High Level Language.

A Brief Introduction to the Internet: The World Wide Web, Web Browsers, Web Servers, Uniform Resource Locators, protocols: Multipurpose Internet Mail Extensions, The Hypertext Transfer

Protocol and Security. Introduction to Operating System: definition, functions, CUI and GUI, different types of OS.

UNIT II **(10 Hrs)**

Number Systems: Base of a number system, Positional number system, Popular number systems(Decimal, Binary, Octal and Hexadecimal), Counting in binary number system, Conversion-Decimal to Binary, Binary to Decimal, Decimal to Octal, Octal to decimal and binary, Decimal to hexadecimal, Hexadecimal to decimal, Binary and octal, Concept of binary addition and subtraction, Complements in binary number systems,1^s Complement, 2^s Complement and their applications, BCD numbers- concept and addition, Concept of parity bit.

UNIT III **(14 Hrs)**

Boolean Algebra and Gate Networks: Logic gates- AND, OR, NOT, NAND and NOR – Truth tables and graphical representation, Basic laws of Boolean Algebra, Simplification of Expressions, De Morgan's theorems, Dual expressions, Canonical expressions, Min terms and Max terms, SOP and POS expressions, Simplification of expression using K-MAP (up to 4 variables), Representation of simplified expressions using NAND/NOR Gates, Don't care conditions, XOR and its applications, parity generator and checker.

UNIT IV **(12 Hrs)**

Sequential and Combinational Logic: Flip flops- Latch, Clocked, RS, JK, T, D and Master slave, Triggering of flip flops, Counters - Synchronous and asynchronous, BCD, Ripple counters, Half adder, Full adder(circuit diagram), Subtractors, Encoders, Decoders, Multiplexers and De-multiplexers, Analog to digital and digital to analog converters

UNIT V: **(12 Hrs)**

The Memory Elements: Concept of Registers, Shift Registers, Flip-flops as building blocks of memory, RAM, ROM and Cache Memory

Books of study:

- M.M.Mano-Digital Logic and Computer design
- P K Sinha- Computer Fundamentals

References:

- Thomas C Bartee- Digital computer Fundamentals
- Floyd- Digital Electronics -

Malvino & Leach- Digital Principles and Applications

SEMSTER I

U1CRCAP2: PROGRAMMING IN ‘C’

Hours/Week: 04 15U1CRCAP02 Hours/Semester: 60

Course Description:

C is a widely used language in systems programming. It’s a language with lot of capabilities. This subject gives an introduction to programming and basic elements of programming like algorithm, flow chart and Pseudo code. The subject starts with the features of C language and basic elements of the language. Programming constructs like if, for, while and do while are dealt with its syntax and applications. Advanced features like functions, arrays, pointers, structures and unions are also dealt here. Pointer being an important concept is dealt with respect to arrays, structures and functions. The concept of files and preprocessors are also introduced. In general, the subject concentrates in all the areas of C programming which is very much helpful for a beginner in Computer Programming.

Objectives:

1. Knowledge and understanding
    - Understand the fundamental programming constructs.
    - Understand and write searching and sorting techniques.
    - Understand a typical C-like program environment.
2. Cognitive skills (thinking and analysis).
    - Be able to understand and analysis any problem and derive its solution.
    - Be able to develop algorithms.
3. Communication skills (personal and academic).
    - Be able to work as a team
4. Practical and subject specific skills (Transferable Skills).
    - Be able to write C-like programs including searching and sorting techniques.

Learning Outcomes:

- Understand the basic terminology used in computer programming
- Write, compile and debug programs in C language.
- Use different data types in a computer program.
- Design programs involving decision structures, loops and functions.
- Explain the difference between call by value and call by reference
- Understand the dynamics of memory by the use of point

- Use different data structures and create/update basic data files.

UNIT I: **(10 Hrs)**

Problem Solving: Problem Definition, Problem Solving, Logic developments tools - Algorithm, Flowcharts, pseudo code, Modular programming, structured and object oriented – top down and bottom up approaches, features of a good computer program.

UNIT II: **(10Hrs)**

C language basics: C character set, Identifiers and keywords, Data types, Enumeration type, constants, variables, declarations, qualifiers – long, short and unsigned declarations, expressions, symbolic constants, input/output functions, compound statements, arithmetic operators, unary operators, relational and logical operators, assignment operators, increment and decrement operators, Precedence and order of evaluation, conditional operators, bit operators, type casting, using library functions in math.h.

UNIT III: **(12 Hrs)**

Control flow: If statement, if….else statement, nested if...else statement, switch statements, looping – for loop, while loop, do … while statements, nested loop structure, break, continue and go to statements.

Arrays & Strings: Single dimensional arrays, multidimensional arrays, initializing array using static declaration, Searching and sorting of Arrays, Array of Characters, Character arrays and strings, String handling Functions.

UNIT IV: **(14 Hrs)**

User Defined Functions: Function declaration, definition & scope, recursion, Arrays and functions, call by value, call by reference, Storage Classes: automatic, external (global), static & registers.

Structures: Definition of Structures, declaration, structure passing to functions, array of structures, arrays with in structures, unions, typedef statements.

UNIT V: **(14 Hrs)**

Pointers: Pointer Definition, pointer arithmetic, array & pointer relationship, pointer to array, pointer to structure. **Files:**Types of C preprocessor directives. Introduction to files, fopen(), fscanf(), fprintf(), getc(), putc(), fclose(), Simple file handling programs. Concept of command-line-arguments.

Books of study:

- Programming in ANSI C 4E , E. BalaGuruswamy, TMH

- Programming in C, Byron S Gottfried, Shaum's Outline series. TMH

References:

- Computer Fundamentals By P K Sinha&PritiSinha Fourth Edition.
- B. Kernighan and D. Ritchie, "The ANSI C Programming Language", PHI

SEMSTER I

PROGRAMMING IN ‘C’ (LAB)

Hours/Week: 04 15U1PRCAP1 Hours/Semester: 60

[There will be two questions: the first from Exercises 3 to 5 and the second from Exercises 6 to 10. Exercises 1 and 2 will be included in the viva]

1. **Familiarization of Computer System and installation:** Demonstration of various units of Computer system, handling of devices, demo on hardware units, Login process, Booting Process, software installation, driver installation, printer installation etc.

2. **Practicing Operating System Commands:** MS-DOS internal &External commands (dir, copy, del, ren, copy con, date, time, chkdsk, mkdir, cd, rmdir, EDIT etc). MS-WINDOWS –using start menu, desk top, task bar, word pad, note pad, file management- creation, copy, delete, moving of files in directories, selecting and executing a program - Demonstration of editing, compiling and executing a C program using a C compiler.

3. **Programs using Basic Constructs:** Fundamental data types, qualifiers- long, short, unsigned, input/output functions – scanf(), printf(), Arithmetic expressions, Evaluation of integer, real and mixed mode arithmetic expressions, truncation effect, type casting, relational and logical expressions, Conditional operators, trigonometric functions- sin(), cos(), tan(), mathematical functions – abs(), sqrt(), round() defined in math.h, printing formatted outputs using width specifier.

4. **Programs using control structures:** if, switch, for, while, do…while, nested structures, break and continue. Sample programs should include printing of Fibonacci numbers, prime numbers, check for Armstrong numbers, summation series – exp(x), sin series etc and verification of result using built in functions, printing pyramid like pattern & other similar patterns using nested loops.

5. **Programs using Arrays:** Array based programs – Creation of array containing prime numbers, matrix addition, matrix multiplication, transpose of a matrix, array sorting, preparing rank lists based on marks, searching of arrays (linear) for finding price of it, and staticinitialization of arrays.

6. **String manipulation programs:** reading strings using %s, gets(), getchar(), copying one string into another, counting number of characters, vowels, words etc, using string handling functions.

7. **User Defined Functions:** Programs using return type functions, void type functions, example program using recursive functions, array sorting program using function with call by reference, function to copy one string into another.

8. **Program using structures:** array of structures, program using structure containing arrays and array of structures. Rank list preparation
9. Simple program using pointers

10. **Program with files:** creating a data file, reading a data file, Simple file program for file creation and file manipulation. Search for record (linear search) and displaying.

SEMSTER II

MICROPROCESSORS & COMPUTER ORGANIZATION

Hours/Week: 04 15U2CRCAP03 Hours/Semester: 60

Course Description

This course introduces the concepts of 8085 and 8086 microprocessors. The subject also deals with the computer organization. The subject gives description about the central processing unit and memory.

Course Objectives:

The course aims

- to explain the concept of 8086 microprocessor
- to introduce addressing methods and instruction sequencing and execution
- to explain the CPU general register and stack organization
- to explain the instruction formats and classification
- to introduce about different types of memory

Learning Outcomes:

By completing this course the student should be able to:

- Understand the organisation of a computer system in terms of its main components
- Understand the detailed operation of a simple microprocessor
- Understand different processor architectures.
- Understand the central processing unit.
- Understand the various memory and memory mapping techniques

UNIT I: **(10 Hrs)**

Functional units of a computer: Basic operational concepts, Bus structure, Addressing methods, Memory locations and addresses, Instructions and instruction sequencing, Instruction execution.

UNIT II: **(14 Hrs)**

Introduction to the concept of 8086 microprocessor: Introduction, Pin-out Diagram, Operating modes, Operation of 8086, Registers, Interrupts, Bus Cycle, Addressing modes, Instructions of 8086

UNIT III: **(14 Hrs)**

Comparison of various Processors: 16 bit, 32 bit, 64 bit processors-Intel 80286, 80386, 80486, Pentium, Pentium Pro, Pentium II, Pentium III and Pentium 4.

UNIT IV: **(12 Hrs)**

Central Processing Unit: General Register Organization, Stack Organization, Instruction Formats, Instruction Classification.

UNIT V: **(10 Hrs)**

Main Memory: Organization of RAM, SRAM, DRAM,, Read Only Memory-ROM,PROM,EROM,EEPROM, Auxiliary memory, Cache memory, Virtual Memory, Memory mapping Techniques.

Books of study:

- B RAM -Fundaments of microprocessors and micro computers
- M M Mano – Computer Architecture
- Advanced microprocessors & peripherals – Architecture Programming & interfacing by A K Ray & K M Bhurchand

References:

- R S. Gaonkar- Micro processor Architecture, Programming and applications with 8085.
- Venugopal and Ravikanth- Introduction to assembly language programming in 8086
- Barry B. Brey- Intel Microprocessors 8086

SEMSTER II

DATA STRUCTURESUSING 'C'

Hours/Week: 04 15U2CRCAP04 Hours/Semester: 60

Course Description:

The subject deals with the representation of data inside a computer. The subject concentrates on different data structures and their applications. The concepts of arrays linked lists, stacks, queues and trees gives an exposure about how operating systems and compilers are handling their data and symbol table. The subject also deals with different sorting techniques and their time and space complexities. The subject deals with some interesting problems like towers of Hanoi, which can be solved using recursion with the help of a stack. To conclude, the subject gives an overall view of the concepts of systems programming.

Objectives:

- To design and implementation of various basic and advanced data structures.
- To improve the logical ability
- To understand the abstract data types stack, queue, de-que and list.
- To understand prefix, infix, and postfix expression formats.
- To use stacks to evaluate postfix expressions.
- To use stacks to convert expressions from infix to postfix.
- To be able to recognize problem properties where stacks, queues, and de-queues are appropriate data structures.

Learning Outcomes:

On completion of the module the student should be able to:

- Understand a variety of techniques for designing algorithms.
- Understand a wide variety of data structures and should be able to use them appropriately to solve problems
- Understand some fundamental algorithms.

UNIT I: **(10 Hrs)**

Concept of Structured data: Data structure definition, Different types and classification of data structures, Operations on Data structures, Arrays – representation of array in the memory, operations on one dimensional and two dimensional arrays, Bubble sort, Selection sort, linear search, binary search, sparse matrix.

UNIT II: **(12 Hrs)**

Stacks and Queues: organization and operation on stacks – Conversion between infix to postfix & prefix representations- Expression Evaluation - Organization and operations on queues-circular queue-multiple stacks and queue - Applications of stacks and queues.

UNIT III: **(16 Hrs)**

Linked list: Memory allocation (dynamic vs. static), concept of dynamic data structures, linked list, need, basic operations and types of linked list, linked list using pointers, insertion and deletion – examples, circular list – doubly linked lists, garbage collection.

UNIT IV: **(10 Hrs)**

Trees: Concept of recursion, definition of - trees, binary trees, strictly binary trees, complete binary tree and Binary search tree, Creation of binary search tree, traversing methods - examples.

UNIT V: **(12 Hrs)**

File organization: File organizations- sequential, random files, linked organization, inverted files, cellular partitioning, hashing function.

Books of study:

- Data Structures through C (A Practical Approach), G.S Baluja DanapatRai& Co.
- Fundamentals of Data Structures, Ellis Horowitz and Sartaj Sajni Galgotia Publications

References:

- Introduction to data structures in C , Ashok N. Kamthane, Person Education
- Theory and Problems of Data Structures, Schaum’s Outline Series, Seymour Lipschutz
- Data structures using c and C++ ,Tanenbaum

SEMSTER II

DATA STRUCTURES USING 'C' (LAB)

Hours/Week: 04 15U2PRCAP2 Hours/Semester: 60

1. Array search and sort – Bubble sort, Selection sort, linear search, binary search, sparse matrix, polynomial addition.
2. Stack implementation, Application of stacks – Conversion of infix expression to postfix, expression evaluation.
3. Queue implementation, Implementation of circular queue.
4. Linked list- implementation, concatenation etc., circular list and doubly linked list implementation, implementation of stacks and queue using linked lists.
5. Creation and traversal of binary search trees.

SEMSTER II

ENVIRONMENTAL SCIENCE

Hours/Week: 05 Hours/Semester: 75

UNIT I: **(2 Hrs)**

The multidisciplinary nature of environmental studies: Definition, scope and importance, Need for public awareness

UNIT II: **(8 Hrs)**

Natural resources: Renewable and non-renewable resources: Forest resources: Use of over exploitation, deforestation, case studies. Timber, mining, dams and their effects on forests and tribal people. **Water resources:** Use and over utilization of surface and ground water, floods, drought, conflicts over water, dams- benefits and problems. **Mineral resources:** Use of exploitation and environmental effects of extracting and using mineral resources, case studies. **Food resources:** World food problems, changes caused by agriculture and overgrazing, effect of modern agricultural fertilizers- pesticides, water logging, salinity, case studies. **Energy resources:** Growing energy needs, renewable and non-renewable energy sources, use of alternate energy sources, case studies. **Land resources:** Land as a resource, land degradation, man induced landslides, soil erosion and desertification.

- Role of an individual in conservation of natural resources
- Equitable use of resources for sustainable lifestyles

UNIT III: **(6 Hrs)**

Ecosystems: Concept of an ecosystem, Structure and function of an ecosystem, Producers, consumers and decomposers, Energy flow in the ecosystem, Ecological succession, Food chains, food webs and ecological pyramids, Introduction , types, characteristic features, structure and function of the following ecosystem:-

- Forest ecosystem
- Grassland ecosystem
- Desert ecosystem
- Aquatic ecosystem (ponds, streams, lakes, rivers, oceans, estuaries)

UNIT IV: **(8 Hrs)**

Biodiversity and its conservation: Introduction- definition: genetic, species and ecosystem diversity,Biographical classification of India, Value of biodiversity: Consumptive use, productive use, social, ethical, aesthetic and option values, Biodiversity at global, national and local level, India as mega-diversity nation, Hot-spots of biodiversity. **Threats of biodiversity:** habitat lose, poaching of wildlife, man- wildlife conflicts, Endangered and endemic species of India. **Conservation of biodiversity:** In-situ and Ex-situ conservation of biodiversity.

UNIT V: **(8 Hrs)**

Environmental Pollution: Definition, Causes, effects and control measures of: Air Pollution, Water pollution, Soil pollution, Marine pollution, Noise pollution, Thermal pollution, Nuclearpollution. **Solid waste management:** Causes, effects and control measures of urban and industrial wastes, Role of an individual in prevention of pollution, Pollution case studies, **Disaster management:** floods, earthquake, cyclone and landslides

UNIT VI:

Social Issues and the Environment: From unsustainable to sustainable development, Urban problem relate to energy, Water conservation, rain water harvesting, water shed management, Resettlement and rehabilitation of people, its problem and concerns.

Case studies:

Environmental ethics: Issues and possible solutions, Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust.

Case studies:

Wasteland reclamation, Consumerism and waste products, Environment Protection Act, Air (Prevention and Control Pollution) Act, Water (prevention and control of Pollution) Act, Wildlife Protection Act, Forest conservation Act, Issues involved in enforcement of environmental legislation, Public awareness.

UNIT VII: **(6 Hrs)**

Human Population and the Environment: Population growth, variation among nations, Population exploitation- Family welfare programme, Environment and Human health, Human rights, Value education, HIV/AIDS, Women and child welfare, Role of information technology in environment and human health, Case studies.

Unit 8: Field work

- Visit to a local area to document environmental assets- river/forest/grassland/hill/ mountains.
- Visit to a local polluted site urban/rural/industrial/agricultural.
- Study of common plants, insects, birds.
- Study of simple ecosystem, pond, river, hill slopes etc.

- Each student has to submit a field report on any one of the above topics which forms the basis for evaluation of field work for 25 marks.

References:

1. Agarwal, K.C.2001 Environmental Biology. NidhiPubl.Ltd.Bikaner.
2. BharuchaErach, The Biodiversity of India. Mapin Publication Pvt.Ltd, Ahamadabad-380013, India , Email: mapin@iccnel.net (R)
3. Brunner R.C, 1989, Hazardous Waste Incineration, McGraw Hill Inc.480p
4. Clark R.S, Marine Pollution, Clanderson Press Oxford (TB)
5. Cunningham, W.P.Cooper, T.H.Gorhani,E& Hepworth, M.T.2001, Environmental Encyclopedia, JaicoPubl House, Mumbai, 1196p
6. De A.K, Environmental Chemistry, Wiley Eastern Ltd.
7. Down to Erath, Centre for Science and Environment (R)
8. Gleick, H.P.1993. Water in crisis, Pacific Institute for studies in Dev, environment & Security. Stockolmenv. Institute. Oxford Univ. Press.473p
9. Hawkins R.E, Encyclopedia of Indian Natural History, Bombay Natural History Society, Bombay (R)
10. Heywood, V.H & Watson , R.T. 1995, Global Biodiversity Assessment, Cambridge Univ.Press1140p
11. Jadhav, H &Bhosale, V.M,1995, Environmental Protection and Laws.Himalaya Pub House, Delhi284p
12. Mckinney, M.L &schoch R.M. 1996. Environmental Science system & Solutions, Web enhanced edition , 639p
13. Mhaskar A.K, Matter Hazardous, Tecno-Science publication(TB)
14. Miller T.G. Jr., Enviromental Science, Wadsworth Publicating Co. (TB)
15. Odum, E.P. 1971. Fundamentals of ecology. W.B. Saunders Co. USA, 574p
16. Rao M.N &Datta, A.K.1987, Waste Water treaement, Ofxord& IBH Publ, Co. Pvt.Ltd.345p
17. Sharma B.K.2001. Environmental Chemistry. Goel Publ. House, Meerut
18. Survey of the Environment. The Hindu (M)
19. Townsend C, Harper J, and Michael Begon, Essentials of Ecology, Blackwell Science (TB)
20. Trivedi R.K, Handbook of Environemental Laws, Rules, Guidelines, Compliances and Standards, Vol I and II, Enviro Media (R)
21. Trivedi R.K and P.K Goel , Introduction to air pollution, Techno-Science Publication (TB)
22. Wagner K.D, 1998. Environmental Management. W.B Saunders Co. Phi;Adelphia, USA 499p

SEMSTER III

DATA COMMUNICATION AND COMPUTER NETWORKS

Hours/Week: 04 15U3CRCAP05 Hours/Semester: 60

Course Description:

This course is to provide students with an overview of the concepts and fundamentals of data communication and computer networks. This course covers layered network models (OSI reference model, TCP/IP networking architecture) and their protocols, wired and wireless networks, mobile computing, error detection and correction and cloud computing and its advantages.

Objectives:

- Build an understanding of the fundamental concepts of computer networking.
- Familiarize the student with the basic taxonomy and terminology of the computer networking area.
- Introduce the student to advanced networking concepts, preparing the student for entry Advanced courses in computer networking.
- Introduce mobile computing and cloud computing.

Learning Outcomes:

After completing this course the student must demonstrate the knowledge and ability to

- independently understand basic computer network technology and its components
- identify the different types of network topologies and protocols
- enumerate the layers of the OSI model and TCP/IP
- understand the characteristics and advantages of mobile computing and cloud computing

UNIT I: (10 Hrs)

Introduction to Data Communication: Components – Data Representation – Data Flow. Networks: Distributed Processing - Network Criteria. **Physical Structures**: Types of Connection. Physical Topology: Categories of Topologies – Bus – Star – Ring – Mesh. Categories of Networks: LAN – MAN - WAN. **Protocols and Standards:** Protocols – Standards - Standards Organizations. **Transmission modes:** Network models – OSI model – seven layers and their functions in OSI model - TCP/IP protocol suite.

UNIT II: **(12 Hrs)**

Data and Signals: Analog and Digital Data – Analog and Digital Signals– Periodic and Non-Periodic Signals. **Periodic Analog Signals:** Sine Wave - Phase - Wave Length –Time and Frequency Domain – Composite Signals – Bandwidth. **Digital Signals:** Bit Rate - Bit Length. **Transmission Impairment:** Attenuation - Distortion –Noise. **Transmission Modes:** Parallel Transmission – Serial Transmission. Multiplexing: FDM – TDM – Synchronous and Statistical TDM – WDM, Spreading, **Transmission Media:** Guided Media –Twisted Pair, Coaxial and Fiber Optic, Unguided Media - Radio Waves – Microwaves – Infrared. **Switching:** Circuit Switching - Datagram Network.

UNIT III: **(12 Hrs)**

Data Link layer: Error detection and Correction: Types of Errors – Redundancy – Detection versus Correction – Forward Error Correction versus Retransmission – Coding – Modular Arithmetic. **Block Coding:** Error Detection – Error Correction – Hamming Distance – Minimum Hamming Distance. Linear Block Codes: Some Linear Block Code. Cyclic Codes: Cyclic Redundancy Check – Checksum. **Framing:** Fixed Size Framing – Variable Size Framing. **Flow Control:** Noiseless Channel Protocol: Simplest Protocol – Stop and Wait Protocol. Noisy Channel Protocols: Stop and Wait ARQ – Go Back N ARQ – Selective Repeat ARQ – Piggy Backing. Multiple Access: **Random Access:** ALOHA – CSMA - CSMA/CD.

UNIT IV: **(14 Hrs)**

Connecting Devices: Hubs, Switches, Repeaters, Bridges, Routers and Gateway. **Network Layer:** Host to Host delivery - Logical Addressing – Internet protocol: IPV4 and IPV6 – Address Mapping – ICMP – IGMP – Unicasting, Multicasting and Broadcasting.

Wired and Wireless LAN: Wireless WAN-Cellular Telephony and Satellite Networks. **Mobile Computing:** Wireless networks: Wireless communication concepts; classification of wireless networks. Cellular networks (1G, 2G, 3G, 4G), WLAN, WPAN, WMAN, Satellite Networks, Mobile and Wireless Devices –Need for Mobile Computing, Mobility management: Handoff and location management concepts.

UNIT V: **(12 Hrs)**

Transport Layer: UDP – TCP, Congestion Control: Open and Closed loop. **Application Layer:** Name Space – Domain Name Space – Label, Domain Name- fully and partially qualified domain names. **Remote logging** - Telnet, FTP, SMTP, and Voice over IP. **Cryptography:** Symmetric and Asymmetric. **Cloud Computing:** cloud computing overview, definition and characteristics, grid computing, difference between grid computing and cloud computing, advantages of cloud computing, cloud service models/types (public, private, hybrid, and community clouds), cloud deployment models (IaaS, PaaS, SaaS, BPaas)

Books for Study and Reference:

- Behrouz and Forouzan - Introduction to Data Communication and Networking - 4th Edition - TMH-2000
- Mobile ComputingTechnology, Applications, and Service Creation by Asoke K Talukder, RoopaYavagal – 1st Edition - McGraw-Hill - 2007
- Cloud Computing By Saurabh K, 2nd Edition - Wiley India Pvt. Ltd.-New Delhi,

SEMSTER III

OBJECT ORIENTED PROGRAMMING IN C++

Hours/Week: 04 15U3CRCAP06 Hours/Semester: 60

Course Description:

C++ is the first object oriented programming language taught in the course. The concept of classes and objects make it easy to represent real world entities. The subject starts with the comparison between procedural languages and object oriented languages. The concepts of operator overloading and function overloading are discussed to expose the students to the advantages of object oriented programming. Advanced topics like inheritance and its various types and virtual functions are taught in depth with its application areas. It is is an important language to learn because of its compact syntax and ability to interact with hardware directly. C++ is used frequently in areas such as game development, hardware manufacturing, embedded systems, and for military applications.

Course Objectives:

- Explain how an existing C++ program works
- Discover errors in a C++ program and describe how to fix them
- Critique a C++ program and describe ways to improve it
- Analyze a problem and construct a C++ program that solves it
- Be able to understand and analysis any problem and derive its solution.

Learning Outcomes:

Upon successful completion of the course, students will be able to

- Know the principles of object-oriented problem solving and programming.
- Outline the essential features and elements of the C++ programming language.
- Explain programming fundamentals, including statement and control flow and recursion.
- Apply the concepts of class, method, constructor, data abstraction, function abstraction, inheritance, overloading, and polymorphism.

UNIT I **(10 Hrs)**

Object Oriented language C++: Basic concept of object oriented programming -benefits of oops-Structure of C++ Program-Basic, derived and user defined data types-Symbolic constants-operators in C++ - Control Structures -Functions in C++-The main function, function prototyping-call by reference-return by reference- inline function-function overloading- friend and virtual functions.

UNIT II: **(12 Hrs)**

classes and objects-specifying a class - Defining member functions - Nesting of member functions - Private member functions - arrays within a class - static data members - static member functions - Arrays of objects-objects as function arguments

UNIT III: **(13 Hrs)**

Constructors and Destructors- Constructors- Parameterized Constructors-Multiple constructors - Copy constructor - Dynamic constructor-Destructors - Operator overloading & Type conversions. **Inheritance**-Defining derived classes-Single, Multiple, Multilevel, Hierarchical and hybrid inheritance- private, public, protected inheritance-virtual base classes-Abstract classes- Constructors in derived classes- nesting of classes.

UNIT IV: **(13 Hrs)**

Pointers-Virtual functions and polymorphism-Pointers-Pointers to objects-this pointer-pointer to derived classes-virtual functions-Pure virtual functions-C++ streams-Stream classes-Unformatted and Formatted console I/O operations- Managing output with manipulators. Manipulation of strings.

UNIT V: **(12 Hrs)**

Exception Handling- Exception Handling, principle of Exception handling, Exception handling mechanism, multiple catch, Nested try, Rethrowing the exception.

Book of study:

- Object Oriented Modeling and Design with UML, Second Edition by James Rumbaugh, Michael Blaha
- Object oriented Programming with C++, Fourth edition By E. Balaguruswamy

References:

- Let Us C++ by YashwantKanetkar - BPB Publications
- Programming with C++ by John R Hubbard - Shaum's Outline series.
- Objected-Oriented Programming in C++ by Rajesh K Shukla, Wiley India.2008
- Edition
- Mastering C++ by Venugopal, Rajkumar, Ravishankar - McGraw Hill

SEMSTER III

SYSTEM ANALYSIS AND DESIGN

Hours/Week: 04 15U3CRCAP07 Hours/Semester: 60

Course Description:

The students of Computer Science always dream to become software professionals. Having learnt a programming language, the students are given an opportunity to know what happens in software development. The students are also exposed to software development life cycle, which basically deals with requirement collection and analysis, Feasibility study and human ware involved in the process. System design, implementation, testing and maintenance are also covered here. The students are asked to analyze the case studies to understand the concepts better. Overall, study of this subject gives a sound knowledge about the software development process.

Objectives:

- understand the principles of systems analysis and design
- be able to carry out a structured analysis of business systems requirements
- be able to design business systems solutions
- understand

Learning Outcomes:

- Understand the principles of systems analysis and design
- Be able to carry out a structured analysis of business systems requirements
- Be able to design business systems solutions.

UNIT I **(12 Hrs)**

Information systems concepts, Business information systems; Describing the business organization – organization chart , organization function list ; information system levels - operational, lower, middle, top management; the system development life cycle concepts; hardware and software end products. Life cycle activities- life cycle flow chart, task, and management review, baseline specifications, role of system analyst.

UNIT II **(14 Hrs)**

Basic tool of system analysis: identification codes – definition, need for codes, code plan, code dictionary, common type of codes, forms design – basic parts of form, style and types of form, principles of form design.

Tools for structure analysis and design: Types of basic charts, decision tables, decision trees, structured English, data flow diagram, data dictionary, system flow charts, flow charting symbols, information oriented flow charts, process-oriented flow charts, HIPO charts.

UNIT III **(12 Hrs)**

Study phase: Study phase activities, information service request, initial investigation, fact finding techniques, fact analysis techniques, steps in feasibility analysis, study phase report.

UNIT IV **(12 Hrs)**

Design phase: Design phase activities, structure design, input design- input data, input media and devices, output design, design phase report, Development phase: Development phase activities, bottom up and top down computer program development.

UNIT V **(10 Hrs)**

Training- programmer, operator, user trainings; conversion; change over plan; PERT; steps in computer program development; structured programming; development phase report.

Software Engineering: Introduction, Role and Nature of Software, Software Terminologies, Role of Management in Software Development. Software Life Cycle Models – Build and Fix Model, Water Fall Model, Prototyping Model, RAD Model, Spiral Model, Iterative Enhancement Model, The Unified Process, Selection of a Life Cycle Model.

Books of study:

- Elements of System Analysis by Marvin Gore & John Stubbe, Galgotia Book Source
- Text book of software engineering by Kumudini Manwar & Manisha Kumbhar

References:

- System Analysis and Design by Elias M Awad, Galgotia Book Source
- Software Engineering Concepts by Richard Fairley, Tata McGraw Publication

SEMSTER III

OBJECT ORIENTED PROGRAMMING IN C++ (LAB)

Hours/Week: 03 15U3CRCAP3 Hours/Semester: 45

1. Basic C++ programs (control structures, functions etc.)
2. Programs using function prototyping, call by reference, return by reference
3. Programs using inline function
4. Programs based on class, objects and manipulation of objects using member functions
5. Programs based on friend functions, passing objects as arguments to function.
6. Programs based on array of objects.
7. Programs based on static data members and static member functions
8. Programs based on function overloading, Default arguments.
9. Programs based on operator overloading (binary, unary) using member functions and friend functions.
10. Programs based on constructors
    - Copy constructor
    - Default constructor.
11. Programs based on Inheritance
    - Single inheritance
    - Multiple inheritance
    - Multilevel inheritance
    - Hierarchical inheritance
    - hybrid inheritance
12. Programs using virtual functions and polymorphism, this pointer
13. Programs using exception Handling

SEMSTER IV

ADVANCED WEB TECHNOLOGY

Hours/Week: 04 15U4CRCAP08 Hours/Semester: 60

Course Description:

This course provides a web development platform on Windows. It allows to create dynamic web applications with HTML5, CSS3, Apache, PHP, and MySQL. This course is designed for the absolute beginner, meaning no experience with WAMP (Windows, Apache, MySQL, PHP).

Objectives:

- To understand the various steps in designing a creative and dynamic website
- To introduce HTML5 and CSS3 using Dreamweaver
- To create good, effective and customized websites
- To practice the hands-on experience in PHP
- To know the data administration using MySQL

Learning Outcomes:

Upon successful completion of the course, students will be able to

- Know regarding internet related technologies. Systematic way of developing a website
- Design dynamic and interactive web pages PHP
- Know the advantages and uses of different types of CSS
- Create powerful database-driven websites
- Demonstrate the ability to author valid externally linked cascading style sheets (CSS)

.

UNIT I **(12 Hrs)**

Introduction to HTML: HTML Tags, creating Forms, creating tables, managing home page. **Introduction to CSS:** Three ways to use CSS Properties, Designing website, working with Templates, Introduction to DHTML.

UNIT II **(14 Hrs)**

HTML5 Introduction: Introducing Dream Weaver, New Elements, Canvas, SVG, Drag/Drop, Geolocation, Video, Audio, Input Types, Form Elements, Form Attributes.

CSS3 Introduction: Syntax, ID& Class, CSS Styling, Styling Backgrounds, Styling Text, Styling Fonts, Styling Links, Styling Lists, Styling Tables, Box Model, Border, Outline, Margin, Padding, Positioning, Floating, Align, Navigation Bar, Image Gallery.

UNIT III **(12 Hrs)**

Introduction to PHP:PHP Basics Syntax, PHP Variables, Global Array and Expression, PHP Operators, PHP Conditional Events and Switch case, PHP Flow Control and Loops, Types of Errors, Array, For each Loop, String Manipulation and Regular Expression, Global Array, String inbuilt functions, Math functions, Array Inbuilt functions.

UNIT IV **(10 Hrs)**

Using HTML Forms: PHP form handling, get data sent from form fields through GET and POST method, form validation, sessions and cookies.

UNIT V **(12 Hrs)**

Introduction to PHP MySQL: Data Types - PHP MySQL Introduction, database structure, tables, MySQL naming rules and columns data types.**PHP MySQL -** INSERT INTO, SELECT, ORDER BY, WHERE and LIKE, UPDATE, DELETE.

Text Books:

- Powell, HTML & XHTM: The Complete Reference, 4th Edition, Tata McGraw-Hill Edition
- Steven Holzner, PHP: The Complete Reference, McGraw-Hill Higher Education, 2008
- Robin Nixon, Learning PHP, My SQL and Java Script, Kindle Edition, OReilly Media 2009.

SEMSTER IV

DATA BASE MANAGEMENT SYSTEM

Hours/Week: 04 15U4CRCAP09 Hours/Semester: 60

Course Description:

All the computer-based applications require data to operate. These data are efficiently handled by a database management system. The subject deals with what is a database and how a database should be designed. It also deals the popular relational data model and SQL queries in depth. It also concentrates on various techniques for database protection and query optimization. The subject also deals with different normalization in brief.

Objectives:

- Understand the terminology and architecture of a database management system
- To give a good formal foundation on the relational model of data
- To present the concepts and techniques relating to query processing by SQL
- Learn the techniques to design an ER model for a database application
- To introduce the concepts of transactions and transaction processing
- Introduce the concept of Data mining

Learning Outcomes:

Upon successful completion of the course, students will be able to

- Identify and define the information that is needed to design a database management system
- Build a database management system that satisfies relational theory with queries, forms, and reports.
- Understand the core terms, concepts and tools of relational database management systems.
- Design entity-relationship diagrams to represent simple database application scenarios

UNIT I: **(12 Hrs)**

Introduction: Characteristics of database approach, Data base users-DBA, Data base designers and end users, Advantages of using DBMS. **Data Modes:** Schemas and instances, DBMS architecture and data independence. **DBMS language:** DDL, DML, DCL Data Base system environment, DBMS Component and modules.

ER Modeling: Introduction- Entity types, Entity sets, Attributes and Keys, Relationship Types, Relationship Sets relationship instances, Constraints on relationship types, Weak entity types, and sample ER diagrams.

UNIT II: (12 Hrs)

Relational Data Model: Relational model concepts domains, attributes, tuples and relations, characteristics of relations. Relational Model constraints Relational Databases and relational data base schemas, entity integrity, referential integrity and foreign keys with examples.

Relational algebra and Relational calculus: Relations Operations, SELECT, PROJECT, UNION, INTERSECTION, The CARTESIAN PRODUCT, JOIN, EQUIJOIN, Aggregate functions. Tuple relations calculus, Domain relationalcalculus.Relational Data base design using ER-to-Relational mapping.

UNIT III: (12 Hrs)

SQL: Data definition commands- CREATE, ALTER,DROP, Adding constraints, Basic SQL queries-INSERT, SELECT,DELETE,UPDATE Ordering of rows UNION,EXCEPT,INTERSET Substring comparisons using LIKE operator, BETWEEN operator, Complex Queries-Nested queries, EXISTS and UNIQUE functions, NULL values, Renaming of attributes and joining of tables, Aggregate functions and grouping, Managing views.

UNIT IV: (14 Hrs)

Data Normalization: Informal Design Guide lines for relation schemas, functional dependencies. **Normal forms:** first, second and third normal form, Boyce- Codd normal form, fourth and fifth Normalisation. **Indexing structures for files:** types of single level ordered indexes.

Transaction processing: Introduction to transaction processing, Transaction and system concepts, Desirable properties of transactions. **Database Security and Authorization:** Types of security, control measures, database security and the DBA, Access protection, User accounts and database audits.

UNIT V: (10 Hrs)

Data Mining Concept: overview of Data mining technology, Association Rules, Classification, commercial data mining tools and applications.

Overview of data warehousing and OLAP: Introduction, definition, characteristics, Building a data Ware House, problems and open issues in Data warehouses.

Books of study:

- Fundamentals of Database Systems by Ramez Elmasri and Shamkant B. Navathe, Pearson Education, 5th edition

References:

- An Introduction to Database systems , C.J Date
- Data base Management Systems by Reghu Ramakrishnan - McGraw Hill International Edition.
- An Introduction to Database Systems by Bipin Desai -Galgoria Publications, 1991

SEMSTER IV

ADVANCED WEB TECHNOLOGY (LAB)

Hours/Week: 02 15U4PRCAP4 Hours/Semester: 30

- Create a vertical frameset with 3 different documents & Jump to a specified section within a frame
- Create a horizontal frameset with 3 different documents & Jump to a specified section within a frame
- Write a HTML program to send a mail from HTML form
- Insert images from another folder or another server in a HTML document and align the image within a text
- Create an application form for a computer center in HTML (use textbox, check box, button)
- Create an advertisement in HTML for a leading software company (animate pictures, link multiple documents)
- PHP Program for the following Date and Time Functions
- Write Java Script program to check for validation in a text box
- Write PHP program to load different advertisement in a web
- Perform the validation in a web application form in PHP
  - Database connection - Bind student database to a DataList control in PHP

SEMSTER V

JAVA PROGRAMMING AND DYNAMIC WEBPAGE DESIGNING

Hours/Week: 04	15U5CRCAP10	Hours/Semester: 60

Course Description:

The course will introduce students to object oriented programming using Java. It assumes that students know the basics of scalar types (integers, strings and booleans) and fundamental control structures in procedural programming (loops, assignment statements, conditional expressions). It will focus on more sophisticated features such as design of classes, interfaces, packages and APIs. It will also cover the basic principles of software design, testing, and collaborative programming. It will finally include a short introduction to the Java Collection Framework and the Java API. This subject deals with Java Programming concepts and dynamic webpage designing using Java Servlets and JSP

Objectives:

- Provide the foundation of good programming skills by discussing keys issues to the design of object-oriented software, including programming design patterns, automatic documentation techniques and programming testing.
- Cover the basics of creating APIs as well as allow students to explore the Java Abstract Programming Interface (API) and Java Collection Framework through programming assignments.
- Build data-driven Web applications with server-side Java technologies
- Add a Web interface to your databases using Java Servlets
- Generate dynamic Web pages with JavaServer Pages (JSP)
- Personalize content for users with cookies and sessions using the Java Servlet API
- Create Web applications using the JavaServerPages framework

Learning Outcomes:

Upon successful completion of the course, students will be able to

- Understand the concept of OOP as well as the purpose and usage principles of inheritance, polymorphism, encapsulation and method overloading.
- Create Java application programs using sound OOP practices (e.g., interfaces and APIs) and proper program structuring (e.g. by using access control identifies, automatic documentation through comments, error exception handling)
- Create object-oriented, scalable, n-tier applications using Java Servlets and Java Server Pages.
- Learn how to integrate key components of the Java Enterprise Edition (Java EE).

- Create dynamic data-driven web applications using servlets and JSP technologies.

UNIT I **(10 Hrs)**

Java Programming: Data types, control structured, arrays, constructors, function overloading, strings, classes, Inheritance, function overriding, Interface, Package, Exception Handling, Multithreaded programming, Priorities of Thread.

UNIT II **(12 Hrs)**

Java applets: AWT controls (Button, Labels, Combo box, list and other Listeners), Layout Manager - Flow Layout, Border Layout, Grid Layout, String Handling (only main functions).

UNIT III **(12 Hrs)**

Networking: Datagram Socket and TCP/IP based server socket, Event handling.

JDBC: Introduction, Drivers, Establishing Connection, Connection Pooling.

UNIT IV **(12 Hrs)**

Java Servlets: Introduction, HTTP Servlet Basics, Servlet Lifecycle, Retrieving Information, Sending HTML Information, Session Tracking, Database Connectivity

UNIT V **(14 Hrs)**

Java Server Pages: Introducing Java Server Pages, JSP Overview, Setting Up the JSP Environment, Generating Dynamic Content, Using Custom Tag Libraries and the JSP Standard Tag Library, Processing Input and Output.

Books for Study and Reference:

- Patrick Naughton and Herbertz Schildt, “Java-2 The Complete Reference” 199, TMH.
- Shelley Powers, “Dynamic Web Publishing” 2nd Ed. Techmedia, 1998.
- Ivor Horton, “Beginning Java-2” SPD Publication
- Jason Hunter, “Java Servlet Programming” O’Reilly
- Shelley Powers, “Dynamic Web Publishing” 2nd Ed. Techmedia, 1998
- Hans Bergsten, “Java Server Pages”, 3 Ed. O’reilly

SEMSTER V

INTERNET, WEB DESIGNING AND CYBER LAWS

(OPEN COURSE)

Hours/Week: 04 15U5CRCAP11 Hours/Semester: 60

Course Description:

This course introduces the concept of the internet, internet services and its applications. The subject also deals with web designing using html, a brief introduction to cyber crimes and cyber laws.

Objectives:

The course aims:

- To explain the basic concepts of internet and internet services
- To explain the facilities for secure communication
- To explain HTML and cyber crimes

Learning Outcomes:

By completing this course the student should be able to:

- Understand the basic concepts related to internet and its standard protocols.
- Design web pages using HTML
- Understand the basic concepts of internet services.
- Understand about E commerce and business
- Understand key terms and concepts in cyber crimes

UNIT I **(10 Hrs)**

Internet – Introduction, Basic Communication, Local Area Network, Packet Switching, Internet: A Network of Networks, ISPs and Network Connections, IP Address, Transmission Control Protocol (TCP), Domain Names.

UNIT II **(10 Hrs)**

Internet Services: Electronic mail, Bulletin Board Service (Network News), browsing the World Wide Web, Automated Web Search (Search Engines), Audio and Video Communication, Faxes and Files (FTP), Remote Login.

UNIT III **(12 Hrs)**

E-Commerce: Facilities for Secure Communication, Electronic Commerce and Business.

UNIT IV **(16 Hrs)**

Web Programming - Introduction to Html, Creating Web Pages, Formatting Tags, Font, lists, table, form, marquee, frame tags, Creation of simple Web Sites.

UNIT V **(12 Hrs)**

Cyber Crimes –Computer Crime, Nature of Crimes, Penalty for damage to Computer, Computer system, tampering with Computer Source Documents, Hacking, Computer Related Offences, Theft, The Language of Cyberspace.

Books of study:

- "The Internet", Douglas E. Comer, Prentice –Hall of India, Third Edition.
- HTML Black Book
- "Cyber Law Crimes", Barkha and U. Rama Mohan, Asia Law House, New Edition.

References:

- "Internet Complete Reference", Harley Hahn

SEMSTER IV

JAVA PROGRAMMING AND DYNAMIC WEBPAGE DESIGNING (LAB)

Hours/Week: 02 15U5PRCAP5 Hours/Semester: 30

1. Using Java (10 Hrs)

- Inheritance
- Interfaces
- Package
- Multiple Inheritance,
- Multithreading
- Exceptions Handling
- Applets
- Layout Managers
- String Functions

2. **Using Servlet** (10 Hrs)

- Write a Servlet to take name as request parameter from the user and display Welcome Page to the user with his name (with Get and Post HTTP methods).
- Write a Servlet with doGet method and extract all the request information along with the inputs given by the user in text field, radio button, checkbox and drop down menu.
- Create 2 HTML pages, one with GET method and one with POST method, with the action to the same Servlet. Handle both the requests using doGet and doPost methods.
- Write a Servlet to extract the multiple values sent with the request by the form with the same name.
- Write a Servlet class to extract names of the request headers, client's browser information, cookies associated, the name of the HTTP method, remote port, local port, server port, content length.
- Write a Servlet class to download a file kept inside a folder within your application, on the request of the user. Take filename as input from the user and if file is not available show the appropriate message.
- Write a Servlet based query system, take query as input from the user in the textfield to extract the data from DataBase and display it to the user in tabular format.

- Write a Servlet program to redirect the request to some other page using sendRedirect method.

3. Using JDBC and JSP (10 Hrs)

- Write a Program using Servlet and JDBC for developing online application for displaying TEE results of MCA Program. A student has to score 50 % in theory, practical and assignment to qualify the paper. Create appropriate databases.
- Write a JSP Program, which displays a web page containing the name of the school, program being offered currently, number of students enrolled in each program, new programs to be offered, eligibility criteria for taking admission in each program.
- Write a program using JDBC and JSP to display the names and addresses of all those MCA students who are working in Software Development Company.
- Develop a web page using Servlet for session tracking which ask for your name and address and print a message of welcome along with the number of times you have visited the page. If you are visiting the page first time the message 'Welcome!'.
- Write a JSP Program, which displays a web page containing two web links. When one click on Link for getting current date and time it goes to a JSP page which display current date and time and by clicking on link for schedule of your practical sessions of MCSL 54 another JSP page will open to show the schedule.
- Write a program using JDBC and JSP to display the names and addresses of all those MCA students at your study center who have completed/submitted their theory assignments of all the courses of IV Semester.

SEMETER VI

SEMSTER VI

OPERATING SYSTEMS

Hours/Week: 05 15U6CRCAP12 Hours/Semester: 75

Course Description:

Operating system is the manager of computer resources. The subject introduces the basic components of an operating system and various types of operating systems. Different CPU scheduling techniques gives a clear picture about how an operating system handles different types of tasks. Memory management policies are dealt to give information about how the main memory can be handled by an operating system. Since the secondary storage devices cannot handle more than one request at a time, it is necessary for an operating system to synchronize the jobs waiting for the devices. There are different types of algorithms, which can be used for the same. The subject deals with all these algorithms, which is of great help for an operating system designer. Case studies about DOS and Windows NT are also dealt as part of this subject.

Objectives:

- To introduce the fundamental concepts and principles of operating systems
- To emphasize the functions of operating systems to the computer system, the system operator (user), and variations in the design and implementations of operating systems.
- To focus on Foundational concepts, Process management, Memory management, virtual memory, file systems and I/O management

Learning Outcomes:

- The course will allow students to understand the fundamental principles for the analysis, design, and development of operating systems – measured by examinations
- Student will be able to identify the major components parts of an OS and able to develop a design schema or architecture
- Students will be able to evaluate or validate the OS principles via simulations and/or realistic
- Students will be able to integrate OS and programming language concepts to solve theoretical problems – such as interrupts or similar mechanisms for synchronization, process management and resource scheduling
- Explain the significance of UNIX operating System

UNIT I **(11 Hrs)**

Introduction: OS Definition, Functions, types of OS, Operating System Operations, Operating System Services, Operating System Interface, System Calls, and Types of System Calls.

UNIT II **(16 Hrs)**

Process: Basic Concepts, Process Scheduling, Operations on 'Processes, Inter process communication, Process Scheduling - Scheduling Criteria, Scheduling Algorithms, Multiple Processor Scheduling.

UNIT III **(17 Hrs)**

Process Coordination: Synchronization - The Critical Section problem, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors. Dead Locks : System Model, Dead Lock Characterization, Methods of Handling Dead Locks, Dead Lock Prevention, Dead Lock Avoidance, Dead Lock Detection, Recovery from Dead Lock.

UNIT IV **(17 Hrs)**

Memory Management: Memory Management Strategies -Swapping, Contiguous memory allocation, Paging, Segmentation. Virtual Memory Management- Demand paging, Page replacement.

Storage Management: File System:- File Concept, Access Methods, Directory Structure, protection, Implementing File Systems:-File System Structure, Allocation Methods.

UNIT V **(14 Hrs)**

Linux - Introduction, Basic Features, Advantages, Kernel, Shell, File System, Commands for files and directories

Books for study:

- Operating System Principles, Seventh Edition, Abraham Silberschatz, Peter Galvin and Greg Gagne, John Wiley
- Operating Systems- By William Stallings
- 'Red Hat Linux Bible' by Cristopher Neyus, Wiley Dreamtech India

References:

- Operating Systems- By Milan Kovic (TMH)

SEMSTER VI

CYBER SECURITY

Hours/Week: 05 15U6CRCAP13 Hours/Semester: 75

Objectives:

- To develop an appreciation for the developments in the field of computers
- Be able to accurately understand the situation of cyber crimes and intellectual property rights
- Be able to understand cybercrime investigation and evaluate various laws related to cyber crimes

UNIT I **(13 Hrs)**

E-Commerce: Defining E-commerce, the revolution, E-Commerce business models and concepts, E-Commerce Infrastructure: E-Commerce Development, Marketing and Security Issues

UNIT II **(15 Hrs)**

Cyber Crime: Definition – History and evolution Types and forms of Cyber Crimes -Malicious Code - Computer Viruses, Computer Worms,Computer Trojans, Web Hacking Foot printing, Port Scanning, E-Shoplifting Web Defacement, Denial of Service Attacks, Manipulating Cookies - Email Hacking: Email Hacking using Packet Sniffers, Email Hacking & Phishing, Email Frauds & Phishing, Email Bombing Email Hijacking - Social Engineering.

UNIT III **(15 Hrs)**

Cyber Crime Investigation: Best Practices for Cyber Crime Investigation: Initialising a Search and Seizure Operation Tracking & Tracing Emails, Recovery of Digital Evidence, Setting up a Cyber Crime Investigation Cell Cyber Forensics: Basic Forensic Principles, Forensic Imaging & Verification, Data Recovery and Analysis.

UNIT IV **(17 Hrs)**

Cyber Law: Case studies USA, UK and India. Cyber terrorism Prevention and detection of Cyber Crime – **Cyber Policing Current statutes in India:** Penalties & Offences under the Information

Technology Act, 2000, Offences under the Indian Penal Code, 1860, Issues relating to investigation and adjudication of Cyber Crimes in India Digital evidence IT act 2000 and other legal provisions

UNIT V **(15 Hrs)**

Intellectual property rights: Intellectual Property Issues and Cyberspace – **The Indian Perspective:** Overview of Intellectual Property related Legislation in India, Copyright law & Cyberspace, Trademark law & Cyberspace, Digital Delivery of Intellectual Property Services.

Books for Study and Reference:

- Computers, Internet and New Technology Laws (A comprehensive reference work with special focus on developments in India) by Karnika Seth
- Cyber Law by Chris Reed

SEMSTER VI

PROJECT WORK

Hours/Week: 05	15U6PRCAP6	Hours/Semester: 75

The BSc Computer Application programme prepares the students to take up positions as Programmers, Systems Analysts, Systems Designers in the field related to computer science and information technology or students may go for higher studies in this area. The students are encouraged to involve themselves completely on the project work in their final semester. It is advised to students to develop their project for solving problems of software industry or any research organization. Doing this will give more exposure to handle real life problems of project development.

This project work is kept in BSc CA program to give the opportunity to develop quality software solution. During the development of the project, the students should involve in all the stages of the software development life cycle (SDLC) like requirements analysis, systems design, software development/coding, testing and documentation, with an overall emphasis on the development of reliable software systems. The primary emphasis of the project work is to understand and gain the knowledge of the principles of software engineering practices, and develops good understanding of SDLC.

The project topic shall be chosen from areas of current day interest using latest packages/ languages running on appropriate platforms, so that the student can be trained to meet the requirements of the Industry. This is an individual project. The students can do project any advanced language which is included in their syllabus.

A project report shall be submitted in hard bound complete in all aspects. For internal evaluation, the progress of the student shall be systematically assessed through various stages of evaluation at periodic intervals.