

FAROOK COLLEGE (AUTONOMOUS)

Farook College PO, Kozhikode-673632

U.G Programme in Computer Science

Under

Choice Based Credit Semester System

SYLLABUS

Core & Open Courses

(2022 Admission Onwards)



Prepared By:

Board of Studies in Computer Science

Farook College (Autonomous)

CERTIFICATE

I hereby certify that the documents attached are the bona fide copies of the syllabus of Core Courses offered to B.Sc. Computer Science programme and Open Courses offered by the Department of Computer Science to be effective from 2022 admission onwards.

Date:
Place: Farook College

Principal

B.SC. COMPUTER SCIENCE PROGRAMME:

The B.Sc. Computer Science Programme of the department of Computer Science at Farook College (Autonomous) started in 1999 and aims to provide the students with state of the art knowledge, general competence, and technical skills on an advanced level needed in academics, industry and research. The students are able to acquire the ability to design solutions or systems for complex problems according to the specifications given. The programme also provides students an orientation to pursue their career in Research and Development.

GRADUATE ATTRIBUTES

GRADUATE ATTRIBUTES (GAS) are qualities and skills that students shall acquire while doing their graduation in Farook College. Graduate attributes include theoretical and practical knowledge, skills, attitudes, societal concerns and values that are expected to be acquired by a graduate through studies at Farook College. The graduate attributes include capabilities that strengthen students' abilities for widening current knowledge base and skills, gaining new knowledge and skills, undertaking future studies, performing well in a chosen career and playing a constructive role as a responsible citizen in the society. Graduate attributes are fostered through meaningful learning experiences made available through the curriculum, the total college experiences and a process of critical and reflective thinking.

GRADUATE ATTRIBUTES OF FAROOK COLLEGE

Disciplinary Knowledge and Competency: Graduates have comprehensive knowledge and understanding of their subject area, the ability to identify the disciplinary thought bases, and is able to interpret the contemporary contexts drawing from related disciplines..

Communication Skills and Digital Literacy: Graduates convey ideas and information effectively to a range of audiences for a variety of purposes and are well prepared for living, learning and working in a digital society.

Research and Analytical Skills: Graduates have a sense of enquiry and capability for identifying problems and designing their research designs and methods suitably; They also manage the research projects and publish their findings.

Critical Thinking and Problem-Solving Skills: Graduates are able to apply critical thinking to a body of knowledge and have the capacity to solve non-familiar

and real life problems by way devising and formulating innovations and improvisations.

Cooperation, Team work , leadership skills and professionalism : Ability to work effectively and respectfully with others from diverse culture and facilitate cooperative or coordinated efforts for institutional and community empowerment; they have the capacity to accept and give constructive feedback and they act with integrity and responsibility; they possess potential to take leadership roles in their chosen involvements with a professional behaviour.

Scientific and reflective thinking: Graduates are able to approach problems and phenomena systematically and analyse and interpret evidences and experiences with logical rigour and and draw conclusions.

Moral and Ethical Values: Graduated are able to embrace moral and ethical values in life and act in consistent with their roles as responsible members of the society.

Employability and Entrepreneurship: Enhancement of employability and entrepreneurial skills. Graduates have the skills including the soft skills required to be employed globally and formulate innovative ideas to face the impending challenges of the times.

Sustainable development: Graduate are aware of the need for environmental friendly and sustainable development.

Independent Lifelong learning: Graduates have internalised the need for revitalisation of knowledge and skills and trained in continuous procuring of knowledge while engaging in job and other responsibilities and businesses.

PROGRAMME EDUCATIONAL OBJECTIVES(PEOs):

The following Programme Educational Objectives are designed based on the department mission.

The PEO's are to facilitate graduating students to

PEO 1: acquire advanced knowledge and expertise necessary for professional practice in Computer Science and its applications for employment, higher studies or research.

PEO 2: attain and practice technical skills to identify, analyze and solve complex problems and issues related to Computer Science.

PEO 3: possess a professional attitude as an individual or a team member with consideration for society, professional ethics, environmental factors and motivation for life-long learning.

PROGRAMME OUTCOMES

The College offers B.Sc. Computer Science Programme since 1999. This Programme aims to equip the students with a broad and scientific understanding of the multiple dimensions of human life and its varied experiences so that they can get hold of current developments and make learned interventions to improve the conditions of life of the humanity and act as a globally responsible citizen.

By the end of the Programme, the Student is able to

PO1: Demonstrate broad knowledge and understanding in the field of Science & Technology.

PO2: Express thoughts and ideas effectively on the knowledge by using appropriate media.

PO3: Analyse and interpret the contemporary developments in the Science around the country and the world.

PO4: Collaborate with persons, institutions or industries who share a unity of purpose and interest in the realm of academics.

PO5: Analyse and synthesize data from a variety of sources and report them in suitable formats.

PO6: Solve or develop solutions for emerging problems/ non familiar problems taking clues from the knowledge acquired through innovations or improvisation.

PO7: Conduct surveys, fieldwork related to research and social outreach and prepare and publish its reports.

PO8: Engage in solving various levels of problems of the people in the community outside the campus.

PO9: Clear the Entrance examinations of renowned HEIs in the country and abroad for further studies.

- PO10:** Use the digital facilities for learning, communication and job related assignments.
- PO11:** Develop aptitudes to find jobs through acquiring soft skills and literary competency and other global competencies.
- PO12:** Imbibe values of love for fellow beings, love for nation, care for others, respect for rights of others.

PROGRAMME SPECIFIC OUTCOMES

By the end of the B.Sc. Computer Science Programme, the Student is able to:

- PSO1:** Demonstrate broad knowledge and understanding in the field of Computer Science and Information Technology like programming, networking, web designing, database systems, operating systems, system software, computer graphics, software engineering etc.
- PSO2:** Express thoughts and ideas effectively on the knowledge in computer science by using appropriate media and technologies like HTML and programming languages.
- PSO3:** Analyse and interpret the contemporary developments in the field of Computer Science/Information Technology around the country and the world through interactions and deliberations.
- PSO4:** Collaborate with persons, institutions or industries who share a unity of purpose and interest in the realm of academics in general and IT/Computer Science in particular through expert talks, industry visits and internships.
- PSO5:** Analyse and synthesize data from a variety of sources and report them in appropriate formats.
- PSO6:** Solve or develop solutions for emerging problems/ non familiar problems taking clues from the knowledge acquired by way of application, innovations or improvisation of existing technologies, techniques or methods.
- PSO7:** Do surveys, fieldwork and Software/ IT projects related to research and social outreach and prepare and publish its reports.
- PSO8:** Engage in solving various levels of problems of the people in the community outside the campus using IT/Computer Technology.

PSO9: Clear the Entrance examinations of renowned HEIs in the country and abroad for further studies/ upward mobility.

PSO10: Use the digital facilities/technologies for learning, communication and job related assignments.

PSO11: Develop aptitudes to find jobs through acquiring soft skills and literary competency and other global competencies in technology related to Computer science/IT.

PSO12: Imbibe values of love for fellow beings, love for nation, care for others, respect for rights of others and of keeping ethics & privacy while using technology.



**B.SC. COMPUTER SCIENCE CURRICULUM STRUCTURE
(2022 - ADMISSION ONWARDS)**

SEMESTER I

Course		Title	C	Mark Weightage			Hours Per Week		
No.	Code			I	E	T	L	P	T
1	A01	Common English Course 1	3	15	60	75	5	0	5
2	A02	Common English Course 2	3	15	60	75	4	0	4
3	A07	Additional Language Course 1	4	20	80	100	5	0	5
4	BCS1B01	Computer Fundamentals	2	15	60	75	1	2	3
5	BMT1C01	Complementary Mathematics 1	3	15	60	75	4	0	4
6	BPH1C01	Complementary Course 1 Physics	2	15	60	75	2	0	2
		Complementary Course 5 Physics Practical					0	2	2
Total (Excluding Audit Course)			17				21	4	25

SEMESTER II

Course		Title	C	Mark Weightage			Hours Per Week		
No.	Code			I	E	T	L	P	T
7	A03	Common English Course 3	4	20	80	100	5	0	5
8	A04	Common English Course 4	4	15	60	75	4	0	4
9	A08	Additional Language Course 2	4	20	80	100	5	0	5
10	BCS2B02	Problem solving using C language	2	15	60	75	1	0	1
11	BCS2B03L	Lab 1: Programming in C Language	2	15	60	75	0	2	2
12	BMT2C02	Complementary Mathematics 2	3	15	60	75	4	0	4
13	BPH2C02	Complementary Course 2 Physics	2	15	60	75	2	0	2
		Complementary Course 5 Physics Practical					0	2	2
Total (Excluding Audit Course)			21				21	4	25

See Legend¹

¹*Credit of Audit courses will not be counted for the calculation of SGPA or CGPA, C: Credits, I: Internal Component, E: External Component, L: Lecture Hours, P: Practical Hours, T: Total Hours

SEMESTER III

Course		Title	C	Mark Weightage			Hours Per Week		
No.	Code			I	E	T	L	P	T
14	BCS3A11	Numerical Skills	4	20	80	100	4	0	4
15	BCS3A12	Web Designing	4	20	80	100	4	0	4
16	BCS3B04	Data Structures	3	15	60	75	2	2	4
17	BCS3B05	Operating System Concepts	3	15	60	75	1	2	3
18	BMT3C03	Complementary Mathematics 3	3	15	60	75	5	0	5
19	BPH3C03	Complementary Course 3 Physics 3	2	15	60	75	3	0	3
		Complementary Course 5 Physics Practical					0	2	2
Total (Excluding Audit Course)			19				19	6	25

SEMESTER IV

Course		Title	C	Mark Weightage			Hours Per Week		
No.	Code			I	E	T	L	P	T
20	BCS4A13	Object Oriented Concepts through Python	4	20	80	100	4	0	4
21	BCS4A14	Principles of Software Engineering	4	20	80	100	4	0	4
22	BCS4B06	Fundamentals of Database Management System and RDBMS	3	15	60	75	3	2	5
23	BCS4B07L	Lab 2: Data Structures and RDBMS	3	15	60	75	0	2	2
24	BMT4C04	Complementary Mathematics 4	3	15	60	75	5	0	5
25	BPH4C04	Complementary Course 4 Physics 4	2	15	60	75	3	0	3
26	BPH4C05L	Complementary Course 5 Physics Practical	4	20	80	100	0	2	2
Total (Excluding Audit Course)			23				19	6	25

SEMESTER V

Course		Title	C	Mark Weightage			Hours Per Week		
No.	Code			I	E	T	L	P	T
27	BCS5B08	Computer Organization and Architecture	4	20	80	100	4	0	4
28	BCS5B09	Java Programming	4	20	80	100	2	4	6
29	BCS5B10	Web Programming using PHP	4	20	80	100	2	4	6
30	BCS5B11	Computer Networks	3	15	60	75	4	0	4
31	BCS5DXX	Open Course	3	15	60	75	3	0	3
		Project Work	0				0	2	2
Total			18				15	10	25

SEMESTER VI

Course		Title	C	Mark Weightage			Hours Per Week		
No.	Code			I	E	T	L	P	T
32	BCS6B12	Computer Graphics	4	20	80	100	3	2	5
33	BCS6B13	Mobile Operating System	4	20	80	100	4	0	4
34	BCS6B14	System Software	3	15	60	75	4	0	4
35	BCS6B15L	Lab 3: Java and PHP	3	15	60	75	0	3	3
36	BCS6B16L	Lab 4 : Android and Shell Scripting	3	15	60	75	0	3	3
37	BCS6EXX	Elective Course	3	15	60	75	4	0	4
38	BCS6B17P	Project Work	2	15	60	75	0	2	2
Total			22				15	10	25

AUDIT COURSES : LIST OF AUDIT COURSES

Course		Title
Semester	Code	
I	AUD01	Environmental Studies
II	AUD02	Disaster Management
III	AUD03	Human Rights
IV	AUD04	Gender Studies

OPEN COURSES : LIST OF OPEN COURSES

Course		Title
No.	Code	
31.1	BCS5D01	Office Automation
31.2	BCS5D02	Online Marketing
31.3	BCS5D03	Content Management System

ELECTIVES : LIST OF ELECTIVE COURSES

Course		Title
No.	Code	
37.1	BCS6E01	Cloud Computing
37.2	BCS6E02	Visual Programming
37.3	BCS6E03	Micro Processor & Micro Controllers
37.4	BCS6E04	Introduction to Internet of Things-IOT

Important notes

Ability Enhancement Courses (Audit Courses):

Ability Enhancement Courses (Audit Courses) are mandatory for the successful completion of the programme but not counted for the calculation of SGPA or CGPA. There shall be one Audit course each in the first four semesters. These courses are not meant for class room study. The students need to attain only pass (Grade P) for these courses. The students can attain these credits through online courses like NPTEL, SWAYAM, MOOC etc.

Extra credit Activities:

Extra credits are mandatory for the programme. Extra credits will be awarded to students who participate in activities like NCC, NSS and Swatch Bharath. Those students who could not join in any of the above activities have to undergo Farook College Social Service Programme (FCSSP). Extra credits are not counted for SGPA or CGPA.

First Semester

BCS1B01: Computer Fundamentals

Course Number: 4

Contact Hours per Week: 3

Number of Credits: 2

Course Description:

This is an introductory course which covers the basic concepts of hardware, software, computer organization, problem solving and C programming language.

Course Outcomes:

- C01** Students will be able to summarize the concepts of software, hardware, programming languages and computer organization
- C02** Students will be able to explain the fundamental concepts of operating systems
- C03** Students will be able to create solutions for simple mathematical and scientific problems systematically
- C04** Students will be able to summarize the characteristics of C language, C program structure and the procedure of writing and executing C programs in various environment
- C05** Students will be able to write sequential C programs to solve simple mathematical and scientific problems.

Course Outline:

Module 1 Concept of Hardware and Software: Computer Languages – Machine Language, Assembly Language, High-level Language, Language translators: Compiler, Interpreter, Assembler, Features of a good language. Basic Computer Organization: Von Neumann model, Input Unit, Output Unit, Storage Unit, Control Unit, memory hierarchy, registers, cache memory, RAM, ROM, PROM and EPROM, Secondary storage devices, storage capacity, bit, byte, nibble

Module 2 Introduction to Operating System: Meaning and Definition, Structure of OS, Types of OS, Functions of OS, Basic DOS commands. Introduction to Linux: Basic commands in Linux such as listing files, viewing contents in files, creating and deleting directories, moving and copying files and/or directories, man pages, setting permissions on files/directories and vi editor. Introduction to Windows OS environment: Creating files/ directory, moving file, printing file

Module 3 Fundamentals of problem solving: The problem-solving aspect, Top-down design, Purpose of programming, the concept of algorithm, flow chart- symbols, sample algorithms- Sum, Average, Finding Smallest Number, Checking Odd/ Even Number, Prime number, Quadratic equation

Module 4 Introduction to C Programming: Overview and importance of C, C Program Structure and Simple programs, Creation and Compilation of C Programs under Linux and Windows Platforms.

Module 5 Elements of C Language and Program constructs: Character Set, C Tokens, Keywords and Identifier, Constants, Variables, Data types, Variable declaration and assignment of values, Symbolic constant definition. C Operators, arithmetic operators, relational operators, and logical operators, assignment operators, increment and decrement operators, conditional operators, special operators, arithmetic expressions, evaluation of expressions, precedence of arithmetic operators, Type conversion in expressions, operator precedence and associativity, Mathematical Functions, I/O operations.

References:

1. Computer Fundamentals by P.K Sinha
2. An introduction to Digital Computer design by V. Rajaraman and T. Radhakrishnan
3. Computer fundamentals by B. Ram
4. Linux in a Nutshell, Ellen Siever, Stephen Figgins, Robert Love, Arnold Robbins, O'Reilly.
5. E. Balaguruswami, Programming in ANSI C.

Second Semester

BCS2B02: Problem solving using C language

Course Number:11

Contact Hours per Week:1

Number of Credits: 2

Course Description:

This course covers various aspects of C programming language including control structures, arrays, functions, pointers, file handling etc.

Course Outcomes:

- C01** Students will be able to summarize about control structures and use them to develop C programs to solve problems
- C02** Students will be able to explain the concept of arrays and pointers in C language and apply them in writing C programs
- C03** Students will be able to illustrate the concepts of modular programming using functions and apply it in developing structured programs.
- C04** Students will be able to use structure, union and dynamic memory allocation in C program development
- C05** Students will be able to explain the ideas of file handling, command line operations and macros in C and use them to write advanced C programs

Course Outline:

Module 1 Decision making, Branching and Looping. Decision making with IF statement, Simple IF statement, If else statement, if else if Ladder, Nesting of If, Switch statement, Conditional operator, Go to statement. Looping: While loop, Do-While, and For Loop, Nesting of loops, jumps in loop, skipping of loops

Module 2 Array- one dimensional array, two dimensional and multidimensional arrays. Strings representation of strings, array of strings, string manipulation functions. Pointers- understanding pointers, accessing the address of a variable, declaring and initializing pointers, accessing a variable through its pointer, pointer expressions. Pointer and arrays, pointer and character string.

Module 3 The Concept of modularization and User defined functions, Multifunction Program, function prototypes, function definition, calling functions, various categories of functions, arguments and argument passing methods, Nesting of functions and recursion, functions and arrays, scope and lifetime of variables in functions. Pointers and functions.

Module 4 Structures & Union - structure definition, giving values to members, structure initialization, comparison of structure variables, arrays of structures, arrays within structures, structures within arrays, structures and functions, Unions. Bit fields. Dynamic memory allocation.

Module 5 Files-defining, opening and closing files, error handling, File manipulation functions, command line operations. Pre-processor directives, Macro substitution directives, simple macros, macros with arguments, nesting of macros, Compiler control directives.

References:

1. E. Balaguruswami, Programming in ANSI C.
2. Brian W. Kernighan & Dennis M. Ritchie, The C Programming Language.
3. Yashavant P. Kenetkar, Let us C.
4. Byran Gotfried, Schaums Outline series Programming with C.
5. Ashok N. Kamthane, Programming in C, Pearson, 2nd Edition.

BCS2B03L: Lab 1- Programming in C Language

Course Number: 12

Contact Hours per Week: 2

Number of Credits: 2

Course Description:

This course aims to make the students capable of writing and executing programs by themselves, familiarize with basic system configurations and operations using operating system commands.

- Learners are expected to use laboratory and practice the problems listed in the program list given below.
- The practical record must contain 25 C programs along with algorithm/flowchart and output
- Operating system experiments need not to be written in the practical record.

Course Outcomes:

CO1: Students will be able to execute various Linux and DOS commands

CO2: Students will be able to develop C programs to solve mathematical and scientific problems

CO3: Implement programs using control structures

CO4: Implement arrays and pointers

CO5: Implement user defined functions and recursion

CO6: Write Programs to illustrate the use of structures, dynamic memory allocation and command line arguments

CO7: Implement file handling using C.

CO8: Apply C programming techniques in biological problems.

Program List:

1. Execute basic DOS commands (DIR, MD, CD, COPY CON, TYPE, DEL, RD, DATE and TIME)
2. Execute basic Linux commands (ls, cp, mv, rm, mkdir, cd, rmdir, cat, chmod man and clear)
3. C program to Implement basic arithmetic operations
4. C program to compute simple and compound interest
5. C program to find mean and standard deviation of three numbers
6. C program to find total energy using $E=mgh+1/2mv^2$
7. C program to find area and circumference of a circle and rectangle
8. C program to print even and odd numbers up to n
9. C program to check whether a given number is prime or not
10. C program to check a given number is Armstrong or not
11. C program to print Fibonacci numbers up to a given number n
12. C program to print the transpose of a matrix
13. C program to swap the values of two variables using function and pointers
14. C program to find sum of digits and reverse of a number
15. C program to print prime numbers with in a given limit
16. C program to print Armstrong numbers up to a given number n
17. C program to print Fibonacci numbers with in a given limit
18. C program to find the roots of a quadratic equation
19. C program to prepare grade card of a student
20. C program to find sum of positive negatives in an array
21. C program to print the count of positives negatives and zeros in an array
22. C program to append two arrays

23. C program to find the smallest and largest element in an array
24. C program to sort an array in ascending order
25. C program to the sum of two matrices
26. C program to find the product of two matrices
27. C program to print the factorial of a number using recursion
28. C program to find the sum of elements in an array using dynamic memory allocation
29. C program to illustrate the use of structure
30. C program to illustrate file operation
31. C program to find sum of two numbers using command line argument
32. Write a C program to calculate the average molecular weight of a given double stranded DNA sequence with n base pairs
33. Write a C program to calculate Body Mass Index value
34. Write a C program to compute the pH value for a given H⁺ ion concentration
35. Program to print the nature of a solution for a given pH value
36. Write a C program to calculate the molecular weight of the given single stranded DNA sequence after checking for phosphorylated (p) / dephosphorylated type
37. Write a C program to compute the percentage of a+t and g+c content of a given DNA sequence
38. Program to print the health status of a person using BMI value
39. Program to print whether the given nucleotide base is purine or pyrimidine.

References:

1. Computer Fundamentals by P.K Sinha
2. An introduction to Digital Computer design by V. Rajaraman and T. Radhakrishnan
3. Computer fundamentals by B. Ram
4. Linux in a Nutshell, Ellen Siever, Stephen Figgins, Robert Love, Arnold Robbins, O'Reilly.
5. E. Balaguruswami, Programming in ANSI C.

Third Semester

BCS3A11: Numerical Skills

Course Number: 16

Contact Hours per Week: 4

Number of Credits: 4

Course Description:

The course is intended to familiarize learners with the fundamental concepts of Statistics, Boolean algebra and digital systems. This course covers representation for Binary, BCD (Binary Coded Decimal) and Floating point numbers and the design of simple combinational and sequential logic circuits.

Course Outcomes:

- CO1:** To summarize the data in a diagrammatic and graphical way, solve basic statistics problems and make possible & appropriate interpretations.
- CO2:** Illustrate decimal, binary, octal, hexadecimal and BCD number systems, perform conversions among them and do the operations - complementation, addition, subtraction, multiplication and division on binary numbers. Illustrate the error detection and correction techniques.
- CO3:** Simplify a given Boolean Function and design a combinational circuit. Implementation of Boolean functions using basic and universal gates.
- CO4:** Summarize the basic digital logic gates and realize basic gates using universal gates.
- CO5:** Design and implement combinational and sequential circuits using Digital Logic Gates - Adders, Decoders, Multiplexers, Flip-flops.

Course Outline:

Module 1 Basic Statistics - definition of statistics, characteristics of statistics, uses, scope and limitation of statistics; Frequency distribution: Classification, discrete and continuous series, formation of frequency distribution. Measures of Central Tendency: Mean, Median, Mode, Geometric mean, Harmonic mean, Meaning and simple problems.

Module 2 Number Systems and Codes, Decimal numbers, Binary numbers, Binary arithmetic, 1's and 2's complements, Octal numbers, Hexadecimal numbers, inter conversions of number systems, Digital codes: Binary coded decimal(BCD), Gray code, ASCII code, error detection and error correction codes, Hamming code.

Module 3 Boolean Algebra: Boolean Algebra: Boolean operations, logic expressions, rules and laws of Boolean algebra, DeMorgan's theorems, minterms, maxterms, SOP and POS form of Boolean expressions for gate network, simplification of Boolean expressions using Boolean algebra and Karnaugh map techniques (up to 4 variables)

Module 4 Logic Gates Positive and negative logic, NOT gate, OR gate, AND gate, NAND gate, NOR gate, EX-OR and EX-NOR gates, Universal gates.

Module 5 Arithmetic and Combinational Logic Circuits Half adder, full adder, parallel binary adder, decoders, BCD to 7-segment decoder, multiplexers and demultiplexers, multiplexer and de-multiplexer trees. Introduction to Sequential Circuits: SR latch, SR flip flop, JK flip flop, Master Slave JK flip flop, D type flip flop, T type flip flop.

References:

1. Fundamentals of Statistics, S.C Gupta, Himalaya Publishing House.
2. Thomas L Floyd, Digital Fundamentals, PHI.
3. Malvino & Leach, Digital Principles & Applications, TMH.

BCS3A12:Web Designing

Course Number: 17

Contact Hours per Week: 4

Number of Credits: 4

Course Description:

The course is intended to make the learners design Web pages by their own.

Course Outcomes:

CO1: Explain the fundamental concepts related to the Internet.

CO2: Develop simple web pages using HTML, CSS and Javascript.

CO3: Design and deploy websites and blogs with the help of Joomla.

Course Outline:

Module 1 Introduction to Internet. What is Internet –Services provided by internet, Applications, Domain Names, DNS, WWW, URL, Browsers, Url, Search Engine, Web Servers. Types of Internet Protocol - HTTP, HTTPS, FTP, Email (POP, IMAP, SMTP).

Module 2 Web page Design using HTML: Introduction, Structure of HTML document, Elements, Attributes, Types of Elements and Attributes, Basic formatting tags: heading, paragraph, underline break, bold, italic, underline, superscript, subscript, font and image. Attributes: - align, color, bgcolor, font face, border, size. Tables: table creation, rows, columns, heading, border. Lists and their Types, Tables.

Module 3 CSS: Cascading Style Sheets- Different types of style sheets-inline, internal, external. syntax, selectors- element, id, class, universal, comments, CSS Background, CSS Border, CSS Text, CSS Font, CSS Images, CSS Tables, CSS Lists

Module 4 Client-Side Scripting using JavaScript: Client-side scripting - Introduction, uses, advantages. Including JavaScript in HTML, variables, comments, output. Operators (Arithmetic, Assignment, Relational, Logical), Conditional statements: if, if else, switch. Control flow statements: while, do while, for, break, continue. Popup boxes: alert, prompt, confirm.

Module 5 CMS: Content Management System. Website development using Joomla: Joomla Basics., Joomla FrontEnd: Understanding Basic Joomla Template. Working with and Creating Content for Joomla: Defining Content, Managing Content, Managing Categories, Menus and Menu Items. Joomla BackEnd: Joomla Global Configuration, Article Manager, Archive Manager, Front page Manager, Section Manager, Category Manager, Menu Manager, Component Manager.

References:

1. HTML 5 Black Book Second Edition, DT Editorial Services, Dreamtech Press, New Delhi.
2. The official Joomla Book, Jennifer Marriott and Elin Waring, Addison Wesley.
3. www.w3schools.com
4. www.html-5-tutorial.com

BCS3B04:Data Structures

Course Number: 18

Contact Hours per Week: 4

Number of Credits: 3

Course Description:

This course aims at molding the learner to understand the various data structures, their organization and operations. It introduces abstract concepts for data organization and manipulation using data structures such as stacks, queues, linked lists, binary trees, heaps and graphs for designing their own data structures to solve practical application problems in various fields of Computer Science. The course helps the learners to assess the applicability of different data structures and associated algorithms for solving real world problems which requires comparing and selecting appropriate data structures to solve the problem efficiently.

Course Outcomes:

- CO1:** Students will be able to Identify suitable data structures and algorithms to solve real world problems. Students will be able to Design an algorithm for a computational task and calculate the time/space complexities of that algorithm. Students will be able to Compare various kinds of searching and sorting techniques
- CO2:** Students will be able to construct Linear and nonlinear data structures using arrays and linked list Students will be able to Develop Programs employing dynamic memory management
- CO3:** Students will be able to Write an algorithm to find the solution of a computational problem by selecting an appropriate data structure (binary tree) to represent a data item to be processed
- CO4:** Students will be able to Write an algorithm to find the solution of a computational problem by selecting an appropriate data structure (graph) to represent a data item to be processed
- CO5:** Students will be able to create hash tables and collision resolution Techniques

Course Outline:

Module 1 Introduction to data structures - need for data structures, definition, categories of data structures, operations; Array: Storage representation of 1D, 2D and multi-dimensional arrays, Operations on arrays-insertion, deletion, searching-linear and binary searching. Sorting- Insertion sort, Bubble sort, Selection sort, Quick sort, Heap sort and Merge sort methods, Implementation, Comparison of different sorting techniques. Sparse matrix: definition and its representation

Module 2 Linked Lists, creation, operations on linked lists- traversal, insertion, and deletion. Doubly linked list (definitions only). Stacks: Stack - Definition, Operation on stack, Implementation using arrays and linked lists, Applications of Stacks, Function Calling, Recursion- direct indirect recursion, Evaluation of arithmetic Expressions, Conversion of Expressions - Prefix, Infix and Postfix expressions. Queues: Definition, Implementations using arrays and linked lists, Circular queue, De-queue, Priority queues, Applications of queues.

Module 3 Trees - Definition, Basic terminology, Binary trees, Representation of binary trees, Sequential representation of binary trees, Linked representation of binary trees, Traversals, Threaded binary tree. Binary Search Trees: Definition, Insertion, Deletion, Traversal and Searching BST, Heap tree: Insertion and deletion (program not needed).

Module 4 Graphs: Graphs Terminologies, Representation of graphs, Adjacency Matrix, Adjacency List, Adjacency Multi-list, Graph search methods (BFS and DFS), Minimal Spanning Tree, Prims Algorithm and Kruskals Algorithms, Shortest path problem, Dijkstra Algorithm

Module 5 Hashing: Different hashing functions, Methods for collision handling.

References:

1. E. Horowitz S. Sahni, Fundamentals of data structures.
2. Alfred V. Aho, Jeffrey D. Ullman, John E. Hopcroft. Data Structures and Algorithms, Addison Wesley
3. Aron M, Tenenbaum, Data Structure Using C and C++
4. E. Balaguruswami. Data Structures Using C.
5. Seymour Lipschutz, Data Structures

BCS3B05:Operating System Concepts

Course Number: 19

Contact Hours per Week: 3

Number of Credits: 3

Course Description:

This course provides a fundamental knowledge about the objectives / functions of Operating Systems. It also covers processes and its life cycle, various memory management and scheduling algorithms and protection of operating system.

Course Outcomes:

- CO1:** Students will be able to Summarize the evolution of Operating System. Students will be able to outline the functions of Operating System
- CO2:** Students will be able to explain the concepts about processes, process synchronization and deadlocks
- CO3:** Students will be able to differentiate various CPU Scheduling algorithms. Students will be able to summarize different File allocation and Free space management methods.
- CO4:** Students will be able differentiate Contiguous and Non-Contiguous memory allocation. Students will be able differentiate Paging and Segmentation.
- CO5:** Students will be able to summarize the various protection and security mechanisms in Operating Systems.

Course Outline:

Module 1 Operating System Objectives and functions: The Evolution of Operating Systems, Serial Processing, Simple batch Systems, Multi Programmed batch Systems, Time Sharing Systems, Parallel Systems, Distributed Systems, and Real time systems

Module 2 Definition of Process, Process States, Process Control Block, Operations on Process, Process Communication, Communication in Client server System, Basic concepts of threads, Concurrency, Principles of Concurrency, Mutual exclusion, The Critical Section Problem, Synchronization Hardware, Semaphores, Classical Problems of synchronization. Dead lock, dead lock prevention, dead lock detection, and dead lock avoidance.

Module 3 CPU Scheduling: Scheduling Criteria, Scheduling algorithms - FCFS, SJF, Priority, RR, Multilevel, Feedback Queue, File System, Functions and organization, Allocation and Free space management

Module 4 Memory Management, Address binding, Logical Vs Physical address space, Dynamic Loading, Dynamic Linking and Shared Libraries, Overlays, Swapping, Contiguous Memory allocation, Paging, Segmentation, Virtual memory, Demand paging, Page replacement, Thrashing.

Module 5 Protection and security: policy and mechanism, authentication, authorization. Mobile OS: Concepts, history, features, architecture, Case studies: Android, UNIX kernel and Microsoft Windows NT (concept).

References:

1. Silberschatz, Galvin, Gagne, Operating System Concepts, John Willey / Sons.
2. Nutt G.J, Operating Systems: A Modern Perspective, Addison Wesley.
3. William Stallings, Operating Systems, Internals and Design Principles, - PHI

Fourth Semester

BCS4A13:Object Oriented Concepts through Python

Course Number: 23

Contact Hours per Week: 4

Number of Credits: 4

Course Description:

The objective of the course is to provide learners an insight into Python programming, and develop programming skills to manage the development of software systems. It covers programming environments, important instructions, data representations, intermediate level features, Object Oriented Programming and file data processing of Python. This course lays the foundation to develop web applications, Machine Learning, and Artificial Intelligence-based applications and tools, Data Science and Data Visualization applications.

Course Outcomes:

- CO1:** Students will be able to explain the fundamental concepts of OOPS
- CO2:** Students will be able to learn execution environment of Python Students will be able to Write, test and debug Python programs
- CO3:** Students will be able to create solutions for simple mathematical and scientific problems systematically Students will be able to illustrate the uses of conditional (if, if-else, if-elif-else and switch-case) and iterative (while and for) statements in Python programs
- CO4:** Students will be able to write programs using functions

CO5: Develop programs by utilizing the modules Lists, Tuples, Sets and Dictionaries in Python. Design object-oriented programs to implement real life problems and their solutions

Course Outline:

Module 1 Introduction to Object-Oriented Programming -basic concepts of OOPS - Object, Class, Inheritance, Polymorphism, Abstraction, Encapsulation. Advantage of OOPs, Object-based programming language.

Module 2 Introduction to python, features, IDLE, Writing and executing python scripts, data type variables-integers, identifiers, comments, variables and assignments, float point types, user inputs, eval function, print function, operators, , operator precedence and associativity

Module 3 Boolean expressions, Simple if statement, if-else statement, compound boolean expressions, nesting, multi way decisions. Loops: The while statement, the for statement, nested loops, break and continue statements, infinite loops.

Module 4 Functions, standard mathematical functions, time functions, random numbers, writing user defined functions, using functions, main function, parameter passing. Using global variables, default parameters, recursion.

Module 5 String and string operations, List- creating list, accessing, updating and editing list, basic list operations. Tuple- creating and accessing tuples in python, basic tuple operations. Dictionary, built in methods to access, update and delete dictionary values. Set and basic operations on a set. Class in Python - Creating classes, attributes, creating instances, accessing instances, destroying objects.

References:

1. Richard L. Halterman, Learning To Program With Python
2. E. Balaguruswamy, Introduction to Computing and Problem Solving Using Python
3. Martin C. Brown, Python: The Complete Reference

BCS4A14:Principles of Software Engineering

Course Number: 24

Contact Hours per Week: 4

Number of Credits: 4

Course Description:

This course provides fundamental knowledge in the Software Development Process. It covers different Requirement Analysis, Software Design, Development and Maintenance Strategies.

Course Outcomes:

- CO1:** Students will be able to compare various software development methodologies and practices.
- CO2:** Students will be able to describe the requirement analysis and prepare a requirement specification for a given software development problem.
- CO3:** Students will be able to describe different design strategies and be able to prepare a design for a given software development problem.
- CO4:** Students will be able to describe different test strategies and be able to prepare a test plan for a given software development problem.
- CO5:** Students will be able to describe the different project management strategies.

Course Outline:

Module 1 The Evolving Role of Software, Software Myths, Software Engineering: A Layered Technology, Software Process Models, The Linear Sequential Model, The Prototyping Model, The RAD Model, Evolutionary Process Models, Agile Process Model, Component-Based Development.

Module 2 Requirement analysis and specification, Understanding the requirement, Requirement modelling, Requirement Specification (SRS), Requirement Analysis and Requirement Elicitation, Requirement Engineering.

Module 3 Software Design-Design Concepts and Design Principle, Architectural Design, Component Level Design, Function Oriented Design, Object Oriented Design, User Interface Design, Web Application Design.

Module 4 Software Coding / Testing- Coding Standard and coding Guidelines, Code Review, Software Documentation, Testing Strategies, Testing Fundamentals, Testing Techniques and Test Case, Black Box Testing, White Box Testing, Testing Process.

Module 5 Software Maintenance and Configuration Management, types of Software Maintenance, Re-Engineering, Reverse Engineering, Forward Engineering, Agile Software Management- SCRUM framework.

References:

1. Pankaj Jalote, An Integrated Approach to Software Engineering, Narosa Pub.
2. Roger S. Pressman, Software Engineering - A Practical Approach, McGraw Hill publication, Eighth edition, 2014
3. Ian Sommerville, Software Engineering, Pearson Education, Tenth edition, 2015.
4. Ian Sommerville, Engineering Software Products: An Introduction to Modern Software Engineering, Pearson Education, First Edition, 2020.

BCS4B06: Fundamentals of Database Management System and RDBMS

Course Number: 25

Contact Hours per Week: 5

Number of Credits: 3

Course Description:

This course provides knowledge about the basic principles of database and database design. It also covers the basics of RDBMS and database manipulation using SQL

Course Outcomes:

- CO1:** Students will be able to compare File systems and Database Management System , outline the different database system concepts and to develop ER Diagrams
- CO2:** Students will be able to explain the Relational Data Model and its uses, design a Relational Database using ER to Relational Mapping
- CO3:** Students will be able to apply normalization on Relational Database Tables
- CO4:** Students will be able to define and manipulate Relational Database using SQL for real life applications
- CO5:** Students will be outline about Transaction Management and Concurrency Control

Course Outline:

Module 1 Database System concepts and applications: Introduction to data bases, File Systems Versus DBMS. Advantages and Disadvantages of using DBMS Approach, Database administrators and users. Data Models, Schemas, and Instances, Types of Data Models, Three Schema Architecture and Data Independence, Database Languages and Interfaces. Conceptual Data Models for Database Design: Entity Relationship Models- Concept of Entity, Entity Sets, Relationship Sets, Attributes, Domains, Constraints, Keys, Strong and Weak Entities. Concepts of EER

Module 2 Relational Data Model: Relations, Domains and Attributes, Tuples, Keys. Integrity Rules, Relational Algebra and Operations, Relational Calculus and Domain Calculus. Relational Database Design using ER to Relational Mapping

Module 3 Relational database design: - Anomalies in a Database, Normalization Theory, Functional Dependencies. First, Second and Third Normal Forms, Relations with more than one Candidate Key, Good and Bad Decompositions, Boyce Codd Normal Form, Multivalued Dependencies and Fourth Normal Form, Join Dependencies and Fifth Normal Form.

Module 4 SQL- Data Definition in SQL: creation, updation, deletion of tables, modifying the structure of tables, renaming, dropping of tables. Constraints. Database Manipulation in SQL: Select command, Logical operators, Range searching, Pattern matching, grouping data from Tables in SQL, GROUP BY, HAVING clauses, Joins - Joining Multiple Tables, Joining a Table to itself. Views -Creation, Renaming the column of a view, destroys view

Module 5 Transaction Management and Concurrency Control thorough MySQL - Transaction: Properties (ACID), states, Commit, Auto-commit, Rollback; Concurrency Control-Lost update problems, Locks, two phase locking.

References:

1. Fundamentals of Database Systems, Elmasri & Navathe, Pearson Education.
2. Database System Concepts Abraham Silberschatz, Henry F Korth, S. Sudarshan, 5th Ed.
3. Introduction to Database Systems, CJ Date, Addison Wesley.
4. Database Management Systems Ramakrishnan McGraw Hill.

BCS4B07L: Lab 2 – Data Structures and RDBMS

Course Number: 26

Contact Hours per Week: 2

Number of Credits: 3

PART A - DATA STRUCTURES

This course aims to provide hands on experience on various data structure concepts and their implementation through C programming language. It also enhances the capability of the students to design and implement solutions to real world mathematical, scientific and biological computational problems.

- Learners are expected to use laboratory and practice the problems listed in the program list given below.
- Experiments are to be done using C Programming language.
- Minimum number of experiments in the Laboratory Record Book should be 15.
- Laboratory Record Book should include Experiment No, Date, Aim of experiment, algorithm, output and result.

Course Outcomes:

CO1: To apply the knowledge of data structures for solving the simple real world computing problems.

CO2: Examine a given Data Structure to determine its space complexity and time complexities of operations.

CO3: To Implement various sorting, searching techniques.

CO4: Design and implement an efficient data structure to represent given data.

CO5: Apply Python programming techniques in biological problems.

Program List:

1. Write a program to perform array insertion
2. Write a program to delete an item from an array.
3. Write a program to merge two arrays.
4. Write a program to represent a sparse matrix.
5. Write a program to implement recursion.
6. Write a program to implement stack using array.
7. Write a program to implement stack using linked list
8. Write a program to implement queue using array
9. Write a program to implement queue using linked list.
10. Write a program to implement circular queue using array
11. Write a program to create a linked list and insert few data items and traverse it
12. Write a program to create a linked list and implement insertion and deletion
13. Write a program to create a linked list and search a particular data item in it
14. Write a program to implement linear search
15. Write a program to implement binary search
16. Write a program to sort a set of numbers using bubble sort algorithm.
17. Write a program to sort a set of numbers using selection sort algorithm
18. Write a program to sort a set of numbers using insertion sort algorithm
19. Write a program to sort a set of numbers using quick sort algorithm
20. Write a program to search a pattern in a string
21. Write a program to Count Letters in DNA
22. Write a program to perform Efficiency Assessment
23. Write a program to translate genes into proteins

24. Write a program to Dot Plots from pair of DNA sequences
25. Write a program to Extend the constructor in class Gene
26. Write a program to Speed up Markov Chain mutation

References:

1. E. Horowitz & S. Sahni, Fundamentals of data structures.
2. Alfred V. Aho, Jeffrey D. Ullman, John E. Hopcroft. Data Structures and Algorithms, Addison Wesley
3. Aron M, Tenenbaum, Data Structure Using C and C++
4. E. Balaguruswamy. Data Structures Using C.
5. Seymour Lipschutz, Data Structures

PART B - RDBMS

The purpose of this course is to use laboratory and practice implementing programs listed here.

- Minimum number of experiments in the Laboratory Record Book should be 15.

Course Outcomes:

- CO1:** Students will be able to implement DDL commands in SQL
- CO2:** Students will be able to implement DML commands in SQL
- CO3:** Students will be able to implement DQL commands in SQL
- CO4:** Students will be able to manipulate constraints in database using SQL
- CO5:** Students will be able to implement SQL Clauses like BETWEEN, IS, IS NOT, IN, EXISTS, NOT EXISTS, GROUP BY, HAVING etc.
- CO6:** Students will be able to use aggregate functions in SQL commands
- CO7:** Students will be able to implement ASSERTION, VIEW and PROCEDURE in SQL

Program List:

1. **Employee Management System:** Create a database COMPANY. Create a table Employee with column Emp_id, Name, Place and Salary. Insert appropriate details.
 - (a) Display the details of all the employees
 - (b) Display the details of employee whose Emp_id is 200
 - (c) Display the name and place of employee whose salary is greater than 2000
 - (d) Display the name and place of employee whose salary is greater than 5000 and less than 1000.
 - (e) Display the details of employees whose place is Feroke
 - (f) Display the details of employee whose place is Malappuram or Calicut
 - (g) Display the details of employees whose place is Feroke or salary greater than 10000

2. **Student management system:** Create table student with columns reg_no, name, place, age and department. Insert appropriate details in it.
 - (a) Display details of all students.
 - (b) Display the names of students whose age is 20.
 - (c) Display name and department of student whose reg_no is 100.
 - (d) Display the details of student whose name is Ramu or Raju (Hint: Use IN).
 - (e) Display the details of student whose name is Aysha or Ahana and department is Maths.
 - (f) Delete the details of students whose age is 20.
 - (g) Delete the details of students whose place is Feroke or age is 18.
 - (h) Update the details place = Calicut and name = Ammu for whose reg_no is 23.
 - (i) Update the place of the student whose name is Ramu to Wayanad.
 - (j) Delete the details of the students whose age between 18 and 20 (Hint: Use BETWEEN).
 - (k) Delete all the details from the Student table.
 - (l) Remove the table student from the database

3. **Employee Management System with constrains:** Create table Employee with columns emp_id, name, place and salary. Insert appropriate values and perform the following

- (a) Set constraint emp_id as PRIMARY KEY and employer name as NOT NULL.
- (b) Display details of all employees.
- (c) Display names of employees whose place is Feroke or Calicut or Manjeri.
- (d) Display details of employee whose emp_id is 200.
- (e) Remove the details of employee whose salary is greater than 5000 and less than 20000.
- (f) Update the place of employee whose emp_id is 200.
- (g) Display the details of all employees in ascending order of their name.
- (h) Display the details of employees whose name begin with 'A' and end with 'U'.
- (i) Display the details of employees in descending order of their salary.
- (j) Delete details of employee whose salary is greater than 5000 and place is Feroke.
- (k) Update the table and set place as NOT NULL.
- (l) Add a new column Age to the table.
- (m) Update the table and set age as 20 for whose emp_id is 200.
- (n) Display the details of employee whose place contain 'nad' as sub string.
- (o) Display the details of employee whose name contain 'a' as second and 'j' as second last letter

4. Customer Management System with ORDER BY: Create a table Customer with column cust_id, name, place, age, salary, job and insert appropriate data

- (a) Set cust_id as PRIMARY KEY.
- (b) Set name and place as NOT NULL.
- (c) Display the details of customer whose age is less than 30 and job is 'Salesman'.
- (d) Update job details of customer whose cust_id is 101.
- (e) Delete details of customer whose place is Calicut and age is 30.
- (f) Display places of all customers without duplication.
- (g) Display the name of customer whose place begin with 'ko' and end with 'm'.
- (h) Display details of customer whose name begin with 'Aj'.
- (i) Display the name of customer whose salary ends with '000'.
- (j) Display the details of customers in ascending order of their salary.

- (k) Display the details of customer in descending order of their name.
- (l) Remove the primary key constraint.
- (m) Remove the NOT NULL constraint.
- (n) Add column 'Address' to the table.
- (o) Modify the datatype of cust_id from int to string.
- (p) Remove the column salary from the table.
- (q) Display the details of customers whose name contains 'b' as the third letter.
- (r) Remove all data in the table without losing the schema

5. **Customer Management System:** Create a table Customer with column cust_id, name, place, phone_no and insert appropriate data

- (a) Update details of place of customer whose cust_id is between 100 and 105.
- (b) Find out the customer belonging to place 'kozhikode' and cust_id between 103 and 105.
- (c) Delete the records whose place is NULL.
- (d) Display all records in ascending order of their name.
- (e) Display the total number of records in the table

6. **Gross Payment Calculation System:** Create a table employee with following fields

Emp_no - Integer, Primary Key

Name - Char(25)

Designation - Char(20)

Department - Char(15)

Basic Pay - Decimal(8,2)

DA - Decimal(8,2)

Insert data in the fields Emp_no, Name, Designation, Department and basic pay with at least 5 records.

- (a) Update DA in the table as 60% of basic pay.
- (b) Add a new column gross to the table and update gross as sum of DA and basic pay.
- (c) Display all details of employees with highest and lowest gross.
- (d) Display name and designation of employees who have DA below average DA.
- (e) Display structure of table (The query syntax is: DESCRIBE Table Name)

7. **Bank Deposit Management System:** Create a table Deposit with the following fields.
Account number - Integer, Primary Key
Name - Char(25)
Amount - Decimal(8.2), NOT NULL
Branch may be either Calicut, Thrissur or Kochi
Insert data in the fields with at least three tuples.
- Display the branches without repeating.
 - Display total deposits in each branch.
 - Display the details of customers who have deposited exactly 10000.
 - Display account number and name of deposits in Calicut branch.
 - Clear all the depositors from the table and then delete the table
8. **Employee Management System with BETWEEN clause:** Create a table employee with columns emp_id, name, place, department, company and salary. Insert appropriate data.
- Display all employee details.
 - Display the details of employee whose name is Ramu.
 - Update the salary of employee whose emp_id is 101 or company is Microsoft.
 - Delete the details of employee whose name is ramu and company is Microsoft.
 - Display the details of employee whose salary is greater than 35000.
 - Display the details of employee whose salary is greater than 20000 and less than 40000.
 - Delete the details of employee whose company name is Apple
9. **WHERE clause demonstration:** Create table Customer with columns c_id, name and place. Create table Loan with columns loan_no and amount. Insert appropriate values.
- Display the details of Customer.
 - Display the details of Loan.
 - Display the details of Customer whose place is 'feroke' and name end with 'e'.
 - Delete the details of customer whose name begins with 'A' and 'D' as the third letter.
 - Update the details of customer whose place is 'feroke'.
 - Display the name of the customer whose c_id is 2.

- (g) Display the details of loan with highest amount.
- (h) Display the details of loan in descending order of the amount.
- (i) Add a column 'Date' in Load.
- (j) Display the details of loan for whose amount is between 10000 and 50000

10. **Aadhar processing using Foreign Key:** Create the following Aadhar and Phone_no tables.

Aadhar (cnum, name, state)

Phone_no (phone_no, carrier, cnum)

Create the above table by property specifying the PRIMARY KEY and FOREIGN KEY and enter at least four tuples for each relation.

- (a) Display details of all aadhar entries.
- (b) Display all phone_no records.
- (c) Display name from Aadhar table and cnum from Pnone_no table.
- (d) display the name and corresponding phon_no.
- (e) Display cnum, name and phone_no of persons whose carrier is idea

11. **Database Backup System using Procedure:** Consider the following tables.

Department (dept_no, dname, location)

Employee (emp_no, emp_name, designation, hire_date, salary, commission, dept_no)

Empbackup (emp_no, emp_name, designation, hire_date, salary, commission, dept_no)

Create the Department and Employee tables by specifying the PRIMARY KEY and FOREIGN KEY and enter at least four tuples for each relation.

- (a) Display the details of department.
- (b) Display the details of employee
- (c) List the names of employees whose name contains substring 'LA'.
- (d) List the details of employees with salary greater than or equal to the average salary in the table Employee
- (e) Create a View which consists of name, designation and hire_date of all salesmen
- (f) Write a PL/SQL procedure to display the emp_no, designation and salary of all employees in the table Employee
- (g) Create a trigger to backup data on table Employee when deleting records
- (h) Display the name of employees who are not working as salesman, analyst or clerk

(i) Display the structure of the table Employee

12. Student Management using Trigger and Procedure: Create the following tables, then insert the appropriate values and answer the following.

Student (ID, name, place, age, DOB)

Studentbackup (ID, name, place, age, DOB)

- (a) Display the details of student whose place is 'Calicut'
- (b) Display the details of students whose name does not begin with 'A' and place begin with 'K'
- (c) Create a trigger which insert data into Studentbackup after insertion into Student table
- (d) Create a procedure to display all details of Student

13. Order Processing database application: Consider the following relations for an order processing database application in a company.

Customer(cust_no int, c_name string, city string)

Order (order_no int, order_date date, cust_no int, order_amt int)

Orderitem(order_no int, item_no int, quantity int)

Item(item_no int, unit_price int)

Shipment (order_no int, warehouse_no int, ship_date date)

Warehouse (warehouse_no int, city string)

Create the above tables by properly specifying the PRIMARY KEYs and FOREIGN KEYs. Enter atleast four tuples for each relevant relation.

- (a) Produce a listing: cname, no_of_orders and avg_amount where the middle column is the total number of orders by the customer and the last column is the average order amount for that customer.
- (b) List the order_no for orders that were shipped from all the warehouse that the company has in a specific city

14. Employee Management using Trigger and Procedure: Create the following tables and insert appropriate values and answer the following

Employee(emp_id, name, place, salary, department)

Employee_backup(emp_id, name, place, salary, department)

- (a) Display details of all employees whose place is 'Feroke'
- (b) Create a View which contains all details of employees whose salary is greater than 5000
- (c) Create a trigger which insert deleted data from Employee into Emp_backup

- (d) Create a procedure which display all details of Employee
 - (e) Create a procedure which display the details of employees whose place is 'Feroke'
15. **Employee Management System with ORDER BY clause:** Create table Employee with columns emp_id, name, place and salary. Insert appropriate values and perform the following
- (a) Display names of employees whose place is starting with 'k'.
 - (b) Display the details of employees in descending order of their salary
 - (c) Update the table and set emp_id as NOT NULL
 - (d) Add a new column Dept to the table
 - (e) Delete all details from the table without losing the schema
16. **Student Management System with UPDATE and DROP commands:** Create table student with columns reg_no, name, place, age and department. Insert appropriate details.
- (a) Set constraint reg_no as PRIMARY KEY and name as NOT NULL
 - (b) Display the details of student whose name is Arun or Dinoy and department is Malayalam
 - (c) Delete the details of students whose age is less than 20
 - (d) Update the details place = Kottayam and name = John for whose reg_no is 13
 - (e) Remove the table student from the database
17. **Book Management System:** Create a table book with bookid, book_name, author, publisher and price. Insert at least 5 records and perform the following.
- (a) Display the book with lowest price published by the ABC publisher
 - (b) Display the different books available
 - (c) Display the details of books in ascending order of their names.
 - (d) Increase the price of books published by "XYZ" by 10%.
 - (e) Delete all books whose authors are either "Balaguruswamy" or "Tanambaum".
18. **Account Management System:** Create a table Deposit with the following fields Acc_num Primary Key, Name, Amount NOT NULL and Branch. Insert data in the fields with at least three tuples.
- (a) Display the branches without repeating

- (b) Display total deposits in each branch
- (c) Display the details of customers who have deposited greater than 10000
- (d) Display account number and name of depositors in ascending order of name
- (e) Clear all the depositors from the table and then delete the table



Fifth Semester

BCS5B08: Computer Organization and Architecture

Course Number: 31

Contact Hours per Week: 4

Number of Credits: 4

Course Description:

The course is prepared to enable the learners of understanding the fundamental architecture of a digital computer. Computer Organization and Architecture course is essential to understand the hardware behind the code and its execution at physical level by interacting with existing memory and I/O structure. It enables the learners to understand the fundamentals about computer system design. AS a result the learner can extend the features of computer organization to detect and solve problems occurring in computer architecture.

Course Outcomes:

CO1: Design sequential circuits - Registers, Counters and Shift Registers.

CO2: Recognize and express the relevance of basic components, I/O organization and memory organization in a digital computer

CO3: Demonstrate the control signals required for the execution of a given instruction

CO4: Illustrate the design of the Arithmetic Logic Unit and explain the usage of registers in it.

CO5: Use algorithms to perform addition and subtraction on binary and floating-point numbers.

CO6: Apply the control logic for a given arithmetic problem

CO7: Recognize the fundamental ideas of parallel processing systems such as pipelining, array and vector processing.

Course Outline:

Module 1 Sequential Circuits: Shift register: serial in - serial out, serial in - parallel out, parallel in - serial out, parallel in-parallel out configurations. Ring counter, Johnson's counter, synchronous counters and asynchronous counters, up/down asynchronous counter.

Module 2 Basic Computer Organization and Design: Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory reference Instructions, Input, Output and Interrupt Design of Basic Computer, Design of Accumulator logic. Micro programmed Control: Control Memory, Address sequencing, Micro program Example. Design of control unit. Processor Organization: general register organization, stack organization, instruction formats, addressing modes, data transfer and manipulation, program control. Computer Arithmetic: Addition, Subtraction, Multiplication, Division algorithms.

Module 3 Input-Output Organization: Peripheral devices, I/O interface, Modes of Transfer, Priority Interrupt, Direct Memory Access, Input-Output Processor, and Serial Communication. I/O Controllers, Asynchronous data transfer, Strobe Control, Handshaking.

Module 4 Memory Organization: Memory Hierarchy, Main memory (RAM/ROM chips), Auxiliary memory, Associative memory, Cache memory, Virtual Memory, Memory Management Hardware, hit/miss ratio, magnetic and optical storage devices.

Module 5 Parallel Processing: Basic Parallel Processing Architecture - Taxonomy-SISD. MISD, SIMD, MIMD structures - CISC Vs RISC - Symmetric Multiprocessors - Cache coherence and MESI protocol - Clusters - Non Uniform Memory Access. Pipelining: Basic Concepts of pipelining, Instruction Pipelining. Hazards, Reservation Tables, Collision, Latency, Dynamic pipeline, Vector processing / Vector processors.

References:

1. Thomas L Floyd, Digital Fundamentals, PHI.
2. M. Morris Mano, Computer System Architecture, PHI.
3. William Stallings, Computer Organization and Architecture, PHI.



BCS5B09:Java Programming

Course Number: 32

Contact Hours per Week: 6

Number of Credits: 4

Course Description:

This course will provide an insight into OOPS concepts using Java programming language. It also help to develop software package for business projects using java programming.

Course Outcomes:

- CO1:** Students will be able to outline the history and the fundamental programming concepts in java.
- CO2:** Students will be able to illustrate various OOPS concepts using java programming.
- CO3:** Students will be able to demonstrate error handling using exceptions in java
- CO4:** Students will be able to demonstrate concurrent execution using threads in java
- CO5:** Students will be able to illustrate networking and database connectivity using JDBC Connectivity in java
- CO6:** Students will be able to develop GUI based applications in Java

Course Outline:

Module 1 Introduction to Java: History, the Java Virtual Machine, Features, Language Components: Lexical units, Data Types, variables, Type Conversions and Casting, Operators, Control Structures - The for Statement, the if Statement, The while and do while Statements, The switch Statement, The break Statement, The continue Statement. Arrays.

Module 2 Basic OOPS Concepts: Defining New Data Type - Class, declaring objects, methods, Constructors, this keyword, finalize. Overloading methods and constructors, argument passing, Passing Objects to Methods, command line arguments, Access controls, static and final. Inheritance, super keyword, method overriding, dynamic method dispatch, Abstract classes, final and inheritance, Interfaces – defining and implementing interfaces. Packages and import statement. Strings.

Module 3 Exceptions, Threads, Input and Output in Java: need for exception handling, try, catch, throw, throws and finally. Developing user defined Exception classes. Stream and Stream classes, byte stream, character stream, predefined streams, reading keyboard inputs, writing to console, print writer class, buffered streams, reading and writing text files. Threads: threads vs. processes, thread states, advantages of using threads, creating threads by extending Thread class, creating threads by implementing Runnable, thread priorities, daemon threads, thread problems, synchronization

Module 4 Networking and Database Connectivity: Networking fundamentals, the client/server model, InetAddress, URLs, sockets, simple client/server example. Introduction to JDBC: The JDBC connectivity model, drivers, connecting to the database, creating a SQL query, the statement interface, executing SQL queries, getting the results, ResultSet interface, executing SQL update, delete queries, error checking and the SQLException class, Meta data, transaction management.

Module 5 Applets, Events and GUI Applications: Introduction to GUI applications - applets - types of applet, applet skeleton, update method, repaint method, Html Applet tag and passing parameter to applet. Event Handling: The Delegation Event Model, Event Classes, Event Listener Interfaces, Introduction to the AWT, overview of the AWT, structure of the AWT, The AWT hierarchy, containers, components, creating frames, working with color, font and font metrics, graphics – point, drawing lines, rectangles, polygon and ovals. AWT Controls – Label, Button, TextField, TextArea, Checkbox, Choice, List, Scroll bar, Layout and Layout Managers, Menu, MenuBar, MenuItem, Checkbox MenuItem

References:

1. Java Complete Reference, Herbert Schildt, Tata McGraw hill edition.2. Patrick N & Schildt H, Java 2 The Complete Reference, Tata McGraw Hill.
2. J2EE Complete Reference, Jim Keogh, Tata McGraw hill edition.

3. Java Enterprise in a nutshell, David Flanagan, Jim Farley, William Crawford & Kris Mangnusson, OReill.
4. E. Balguruswamy, Programming in Java.



BCS5B10:Web Programming using PHP

Course Number: 33

Contact Hours per Week: 6

Number of Credits: 4

Course Description:

This course introduces client side scripting and server side scripting, which will help the learners to develop their own web applications.

Course Outcomes:

CO1: Learners will be able to design HTML forms for their website

CO2: Learners will be able to apply CSS into their websites.

CO3: Learners will be able to do develop Javascript user defined functions for form validation.

CO4: Learners will be able to illustrate the working of server side scripting.

CO5: Learners will be able to develop basic PHP scripts for transferring data from HTML form to web server.

CO6: Learners will be able to Develop basic PHP scripts for interacting with MySQL database and display the content on a HTML page

Course Outline:

Module 1 HTML5 Form controls: form, text, password, textarea, tel, url, email, range, number, button, submit, reset, checkbox, radio button, select box, data-list, output, hidden, color, date (min and max attributes). CSS: Cascading Style Sheet, types, syntax, selector. Background - Background Color, Background Image. Text - Text Color, Text Alignment, Text Decoration. CSS Lists - Different List Item Markers, Unordered List, Ordered List.

Module 2 JavaScript: Including JavaScript-internal and external, Arrays: Creating, Accessing, for... In loop. Writing JavaScript in HTML head and body elements. Built-in Global Functions: alert(), prompt(), confirm(), isNaN(), Number(), parseInt(). User Defined Functions. Document Object Model (Concept), Events Familiarization: Form Events - onLoad, onunload, onClick, onBlur, onSubmit, onerror, oninput, onReset. Keyboard Events: onkeydown, onkeyup, onkeypress. Form Data Validation using form controls text, password, text area, tel, number.

Module 3 Server side scripting using PHP: Introduction to Server Side Script, PHP, Role of Web server, LAMP / WAMP. PHP Syntax, data types, variables, operators, conditional statements, loops, including files, comments, variables and scope, echo and print.

Module 4 PHP String Manipulation: String Variables in PHP, The Concatenation Operator, String functions - strlen, strpos, strstr, strcmp, substr, str_replace, string case. Working with array: Array constructs - array(), list() and foreach(). User-Defined Functions. Passing information between pages: HTTP GET and POST method. Working with Forms: \$_GET and \$_POST, Form Data Processing at Server side.

Module 5 PHP & MySQL: Features of MySQL, data types, Introduction to SQL commands - CREATE TABLE, SELECT, DELETE, UPDATE, INSERT. PHP functions for MySQL operations: mysqli_connect(), mysqli_select_db(), mysqli_query(), mysqli_fetch_row(), mysqli_fetch_array(), mysqli_fetch_object(), mysqli_close(). Insertion and Deletion of data from MySQL table using PHP, Displaying data from MYSQL in webpage.

References:

1. HTML 5 Blackbook, Dreamtech Press, ISBN 987-93-5119-907-6, 2016 Edition.
2. Jon Duckett, Web Programming with HTML, XHTML, CSS, Wrox Beginning.
3. Jim Converse & Joyce Park, PHP & MySQL Bible, Wiley.
4. Deitel, Harvey M. and Paul J., INTERNET & WORLD WIDE WEB HOW TO PROGRAM,3/E, 2004
5. Beginning PHP, D W Mercer, A Kent, S D Nowicki Publisher: Wrox.

BCS5B11:Computer Networks

Course Number: 34

Contact Hours per Week: 4

Number of Credits: 3

Course Description:

The course introduces main concepts of networking, application areas; classification, reference models, transmission environment, technologies, routing algorithms; IP, UDP and TCP protocols, reliable data transferring methods, application protocols, network security, management systems, perspectives of communication networks

Course Outcomes:

- CO1:** Students will be able to summarize about the fundamental concepts of computer networks, reference models and functions of physical layer
- CO2:** Students will be able to explain the structure and functions of data link layer
- CO3:** Students will be able to outline the working of network layer protocols and its important functions.
- CO4:** Students will be able to explain the architecture and functions of transport layer, transport layer protocols and services.
- CO5:** Students will be able to summarize fundamental application layer protocols, their architecture working and functions

Course Outline:

Module 1 Introduction to Computer networks, categories of networks, Topology, Transmission media - wired and wireless media. Analog and digital data, composite signals, digital data transmission. Transmission Modes- simplex, half duplex, full duplex, serial and parallel, synchronous, asynchronous, Multiplexing - Frequency division multiplexing, Time Division Multiplexing and Wave Division Multiplexing, Switching - Circuit switching, Packet Switching and Message Switching, DTE - DCE Interface. Network Models, Layered model, OSI and TCP/IP models

Module 2 Data link layer, Error detection and correction, Types of errors, Single bit error and Burst error, Vertical redundancy check(VRC), longitudinal redundancy Check(LRC), Cyclic Redundancy Check(CRC), Error correction - Single bit error correction, Hamming code Data compression - Huffman code, data link control, Line discipline, Flow control, Error control, Multiple Access, Random Access, ALOHA, pure ALOHA and slotted ALOHA, CSMA/CD and SCMA/CA, Polling, Wired LANs, Ethernet - IEEE standards, Wireless LANs - IEEE - 802.11, Bluetooth.

Module 3 Network layer, Networking and Internetworking devices - Repeaters, Bridges, Routers, Gateways, Logical addressing - IPv4 & IPv6 addresses, Network Address Translation (NAT), Internet protocols, internetworking, Datagram, Transition from IPv4 to IPv6, Address Mapping- Error reporting and multicasting - Delivery, Forwarding and Routing algorithms, Distance Vector Routing, Link State Routing, Multicast routing protocols, The Dijkstra Algorithm.

Module 4 Transport layer, Process-to-process Delivery: Connectionless and Connection oriented Protocols, User Datagram Protocols, Transmission Control Protocol and SCTP, Congestion control and Quality of Service.

Module 5 Application Layer: Domain Name Systems, WWW, HTTP, File Transfer Protocol (FTP), Trivial File Transfer Protocol (TFTP), Simple Mail Transfer Protocol (SMTP) and Post Office Protocol (POP). Network management: SNMP, Network security, Cryptography.

References:

1. Computer Networks, Fifth Edition by Andrew S. Tanenbaum and David J Wetheral, Pearson Education
2. Data Communications and Networking, Fourth Edition by Behrouz A Forouzan, McGraw-Hill reprint, 2011

Sixth Semester

BCS6B12:Computer Graphics

Course Number: 36

Contact Hours per Week: 5

Number of Credits: 4

Course Description:

This course introduces fundamentals of 2D computer graphics and GIMP.

Course Outcomes:

CO1: Learners will be able to compare various graphic devices.

CO2: Learners will be able to analyze and implement algorithms for line drawing, circle drawing, and polygon filling.

CO3: Learners will be able to apply geometrical transformations on 2D objects.

CO4: Learners will be able to analyze and implement algorithms for clipping

CO5: Learners will be able to differentiate various color models

CO5: Learners will be able to do basic image manipulation using GIMP

Course Outline:

Module 1 Introduction to Computer Graphics Definition, Application, Pixel, Frame Buer, Raster and Random Scan display, Display devices CRT, Color CRT Monitors, basics of LCD & LED Monitors.

Module 2 Scan Conversion of line DDA algorithm of line drawing, Scan conversion of midpoint circle generating algorithm, Polygon Filling, Scan line polygon filling algorithm.

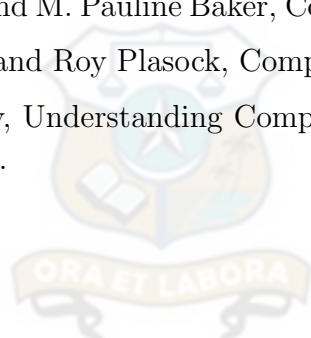
Module 3 Two-Dimensional transformation, Translation, Rotation, Scaling, Homogeneous Coordinates, Reflection, Shear.

Module 4 Window to view port transformation, clipping, line clipping, Cohen Sutherland line clipping, Polygon clipping, Sutherland and Gary Hodgman polygon clipping algorithm.

Module 5 Color Models & Color Applications Light and Color, Different color models, RGB, CMY, YIQ. Introduction to GIMP Image Manipulation using GIMP.

References:

1. Donald Hearn and M. Pauline Baker, Computer Graphics, PHI, New Delhi.
2. Zhigang Xiang and Roy Plasock, Computer Graphics, Schaum's Outlines.
3. Deborah Morley, Understanding Computers Today and Tomorrow, Introductory Edition.



BCS6B13:Mobile Operating System

Course Number: 37

Contact Hours per Week: 4

Number of Credits: 4

Course Description:

The aim of this course is to provide an insight to mobile application development using the Android platform.

Course Outcomes:

- CO1:** Students will be able to explain the android environment, installation, architecture, fundamental components, life cycle etc.
- CO2:** Students will be able to explain the resources and its management in android.
- CO3:** Students will be able to create screen using various UI elements and organize it Students will be able to create simple android applications
- CO4:** Students will be able to create advanced GUI using fragments, dialogs, and action bar.
- CO5:** Students will be able to explain various ways store and retain data in applications Develop applications that communicate with Files, SQLite databases

Course Outline:

Module 1 Introducing the android computing platform, History of android, android software stack, Dalvik virtual machine, Developing end user application using android SDK, android java packages, Setting up the development environment, Installing android development tools (ADT), Fundamental components, Android virtual devices, Running on real device, Structure of android application, Application life cycle. Manifests.

Module 2 Understanding android resources - Enumerating key android resources, String resources - string arrays, plurals, Layout resources, Colour resources, dimension resources, image resources, style, Resource reference syntax. Understanding intents - basics of intents, types of intent, available intents, exploring intent composition, pending intents.

Module 3 User interfaces development in android - building UI completely in code, UI using XML, UI in XML with code, Android's common controls - Text controls, button controls, checkbox control, radio button controls, image view, date and time controls, map view control, understanding adapters, adapter views, list view, grid view, spinner control, gallery control, Understanding layout managers - linear layout manager, table layout manager, relative layout manager, frame layout manager, grid layout manager. Android menus - creating menus, working with menu groups, responding to menu items, icon menu, sub menu,, dynamic menus, loading menu through XML.

Module 4 Fragments in android - structure of fragment, fragment lifecycle, fragment transaction, fragment manager, Using dialogs in android, alert dialog, dialog fragments, working with toast. Implementing action bar – creating action bar, types – standard action bar, tabbed action bar activity, tabbed listener, list navigation action bar activity, list listener, action bar and menu interaction, action bar and search view, action bar and fragments. Introduction to services, bound and unbound service, life cycle of service

Module 5 Persisting data - Files, saving state and preferences – saving application data, creating, saving and retrieving shared preferences, preference activity, preference layout in XML, native preference controls, preference fragments, including static files as resources, Working with file system, SQLite - SQLite types, database manipulation using SQLite, SQL and database centric data model for android, android database classes. Understanding content providers, content URIs, built in providers. Reading, writing and updating data using content URIs.

References:

1. Pro Android 4, SatyaKomatineni & Dave MacLean, Apress.
2. Professional Android 4 Application Development, Retomeier, Wrox.
3. Programming Android, ZigurdMednieks, Laird Dornin, G. Blake Meike, and Masumi Nakamura, O'Reilly

BCS6B14: System Software

Course Number: 38

Contact Hours per Week: 4

Number of Credits: 3

Course Description:

The course introduces the fundamental concepts of system software. This is followed by a discussion on assembler, macros, loaders, linkers and compilers. The course concludes with a case study highlighting the features of LEX (a lexer) and YACC (a parser).

Course Outcomes:

- CO1:** Students will be able to summarize the fundamental concepts of system software.
- CO2:** Students will be able to differentiate between system software and application software.
- CO3:** Students will be able to outline the phases of Assembler.
- CO4:** Students will be able to explain the functions of a macro processor.
- CO5:** Students will be able to summarize the roles taken by loaders and linkers.
- CO6:** Students will be able to explain the structure and functions of compilers.
- CO7:** Students will be able to explain the salient features of LEX and YACC.

Course Outline:

Module 1 System software: General concept, Assemblers, loaders, linkers, macros, compilers, interpreters, operating system, Design of assemblers.

Module 2 Macros and macro processors, Macro definitions and instructions, Macro calls, Features of Macros.

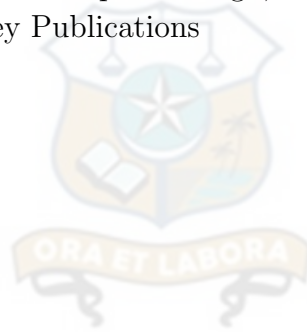
Module 3 Loading, linking and relocating Loader schemes- Binders, linking loaders, overlays, dynamic binders-Dynamic loading and dynamic linking – Relocatability of programs.

Module 4 Compilers - Phases of a compiler - Lexical, Syntax, Intermediate code generation, Optimization, Code generation, Symbol table and error correcting routines – Passes of a compiler.

Module 5 Case studies of lexical and syntax analysers: LEX and YAAC.

References:

1. D.M. Dhamdhere, Systems Programming and Operating Systems
2. John J Donovan, Systems programming
3. Jim Welsh and R M Mckeag, Structured System Programming, Prentice Hall. Principal of Compiler Design, Alfred Aho V and Jeffrey D Ullman, Addison - Wesley Publications



BCS4B15L: Lab 3 – Java and PHP

Course Number: 39

Contact Hours per Week: 3

Number of Credits: 3

PART A - JAVA PROGRAMMING

This course will provide an insight into OOPS concepts by implementing OOPS concepts using Java programming language. It also help to develop software package for business projects using java programming.

- Learners are expected to use laboratory and practice the problems listed in the program list given below.
- Experiments are to be done using Java Programming language.
- Lab cycle consists of 20 experiments.
- Minimum number of experiments in the Laboratory Record Book should be 15.

Course Outcomes:

- CO1:** Students will be able to implement and find solutions to basic problems like checking whether odd or even, prime numbers, Fibinacci series, armstrong numbers etc. using java programming
- CO2:** Students will be able to implement recursion using java programming
- CO3:** Students will be able to implement 1D and 2D arrays using java programming
- CO4:** Students will be able to implement the basic OOPS concepts like class, object, constructors etc using java programming
- CO5:** Students will be able to implement the OOPS concepts like inheritance, overloading, overriding, interfaces etc. using java programming
- CO6:** Students will be able to implement concepts like threads, packages, client-server communication etc. using java programming
- CO7:** Students will be able to implement the concept of error handling using exceptions in java programming
- CO8:** Apply Java programming techniques in biological problems.

Program List:

1. Write a program to check whether a number is odd or even
2. Write a program to print the prime numbers within a range
3. Write a program to print the Fibonacci series up to a limit
4. Write a program to check whether a number is Armstrong or not
5. Write a program to read 3 marks, find its average and provide grades as follows using switch statement (90% - A grade, 80% - B grade, 70% - C grade, 60% - D grade, 50% - E grade, all the rest - failed)
6. Write a program to find the factorial of a number using recursion
7. Write a program to sort an array
8. Write a program to find the transpose of a matrix
9. Write a program to find the roots of a quadratic equation
10. Write a program to append two arrays.
11. Write a program to add two complex numbers
12. Write a program to implement inheritance – Create a class 'Employee' with data members - Name, Age and Basic pay. Create a subclass 'Programmer' with data members - Department and Bonus. And also a method TotalSalary() which calculates the total salary(Basic pay + Bonus) . In main method, create object for only the subclass and display the following details - Name, Age, Department, Total Salary
13. Write a program to implement multi-level inheritance. Create a class 'Person' with attributes Name and age. Inherit a class 'Employee' from 'Person'. 'Employee' must contain the attributes Employee id and Basic salary. Inherit a class 'Manager' from 'Employee'. 'Manager' must contain the attributes Department and salary bonus. Now, create object, only for the 'Manager' class and display all the details about a person.
14. Write a program to find the area of equilateral, isosceles and scalene triangle using method overloading
15. Write a program to compare the area of rectangles using objects as parameters

16. Write a program to create an array of objects (ie. Create a Student class with roll no., name, mark1, mark2 and mark3). Now find the total marks and average marks of each student)
17. Write a program to override a method 'greatest()' for finding the greatest of two numbers and greatest of three numbers
18. Write a program to find the area of a circle and sphere using interfaces
19. Write a program to print 'hai' and 'hello' using threads
20. Write a program to throw an ArithmeticException if the value of a variable 'k' is less than 10
21. Write a program to implement Packages
22. Write a program to implement simple client server application
23. Write a program to find the BMI and BMI Category.
24. Write a program to identify the DNA Symbols.
25. Write a program to print the DNA Sequence.
26. Write a program to input a string as command line argument and print it word by word.
27. Create two interfaces Calcarea which has a method area() and CalcVolume which has a method volume(). Create a class Sphere which implements Calcarea and CalcVolume to find the area and volume of a sphere.
28. Write a program to implement JDBC Connectivity

References:

1. Java Complete Reference, Herbert Schildt, Tata McGraw hill edition.
2. Patrick N & Schildt H, Java 2 The Complete Reference, Tata Mc- Graw Hill.
3. J2EE Complete Reference, Jim Keogh, Tata McGraw hill edition.
4. Java Enterprise in a nutshell, David Flanagan, Jim Farley, William Crawford & Kris Mangnusson, OReill.
4. E. Balguruswamy, Programming in Java.

PART B - Web Programming using PHP

The purpose of this course is to use laboratory and practice web programming with the help of HTML, CSS, JavaScript and PHP.

- Learners are expected to use laboratory and practice the problems listed in the program list given below.
- Minimum number of experiments in the Laboratory Record Book should be 20.

Course Outcomes:

CO1: Create Static web pages with the help of HTML and CSS

CO2: Implement Client side scripting using JavaScript

CO3: Implement Server side scripting using PHP

CO4: Interact with database through PHP

CO5: Design and develop dynamic web pages

Program List:

HTML and CSS

1. Design a webpage that illustrates the use of the following form controls:
 - (i) input controls: single-line text, password, multi-line text.
 - (ii) buttons: submit and reset.
2. Design a webpage that illustrates the use of the following form controls: (i) input controls: check box, radio button, select box (ii) buttons: submit and reset.
3. Design a webpage that illustrates the use of the following form controls: (i) input controls: datalist, multi-select box, grouped select box (ii) buttons: submit and reset.
4. Design a webpage that illustrates the use of field sets and legends.
5. Design a web page to demonstrate Border colors using internal CSS.

6. Design a web page to demonstrate Text alignment using CSS.
7. Design a web page to demonstrate inline CSS.
8. Design a webpage to invert the behavior of the `h1` to `h6` tags using external CSS.
9. Design a webpage for a simple image gallery.

JavaScript

10. Write a javascript program to perform find the area and circumference of a circle.
11. Write a javascript program to check whether the given sides can form a triangle. If yes, find the type (isosceles, equilateral and scalene) and area of the triangle. Use prompt dialogue box to accept the sides.
12. Design a JavaScript program to illustrate the following events: `onLoad`, `onClick`, `onBlur`, `onSubmit`, `onChange`.
13. Design a JavaScript program to display the multiplication table of a no accepted from the user.
14. Design a form that accepts two integers. Provide 4 buttons for Add, Subtract, Multiply, Divide. Add JavaScript program to add, subtract, multiply and divide the given numbers when these buttons are clicked
15. Write a JavaScript program to store different colors in an array and change the background color of the page using this array elements
16. Write a JavaScript program to create clock with a timing event.
17. Write a JavaScript program to perform form validation for question number 1, 2 and 3.

PHP

18. Write a PHP program to check whether the given number is Armstrong or not.
19. Write a PHP program to check whether a given number is perfect, abundant or deficient.
20. Display the Fibonacci series up to a given number.
21. Create a php program to display the bio data of a person by reading the personal details using an HTML page.

22. Create a login page using database.
23. Create a mysql table student with fields roll no, name, mark, grade. Insert records in the table. Write a PHP program to display the mark list of a student by accepting the register no of the student.
24. Write a php application to generate the pay slip of an employee by accepting name, basic salary and designation. The net salary will be calculated based on the following conditions.
- Manager will get 1000 Conveyance allowance and 500 Extra allowance
Supervisor will get 750 Conveyance allowance and 200 Extra allowance
Clerk will get 500 Conveyance allowance and 100 Extra allowance
Peon will get 250 Conveyance allowance
HRA is 25% for all employees
Gross= basic + HRA + conveyance + extra
Net = gross – income tax
Income Tax
 $Gross \leq 2000$: No tax
 $2500 < Gross \leq 4000$: 3% tax
 $4000 < Gross \leq 5000$: 5% tax
 $Gross > 5000$: 8% tax
25. Create a table “product” with fields itemcode, itemname, unitprice. Write php program to insert 5 records into the table and display it in a table format.
26. Write a php program for delete and update operation on account table. The account table contain fields such as accountno, name and amount.
27. Write an HTML page to display a list of fruits in a list box. Write php program to display the names of the fruits which are selected by the user
28. Write a php program to create an array and store 10 names in the array. Do the following operations.
- Display the contents using for each statement.
 - Display the array in a sorted order.
 - Display the array without the duplicate elements
 - Remove the last element and display
 - Display the array in reverse order
 - Search an element in the given array.

References:

1. HTML 5 Blackbook, Dreamtech Press, ISBN 987-93-5119-907-6, 2016 Edition.
2. Jon Duckett, Web Programming with HTML, XHTML, CSS, Wrox Beginning.
3. Jim Converse & Joyce Park, PHP & MySQL Bible, Wiley.
4. Deitel, Harvey M. and Paul J., INTERNET & WORLD WIDE WEB HOW TO PROGRAM,3/E, 2004
5. Beginning PHP, D W Mercer, A Kent, S D Nowicki Publisher: Wrox.



BCS4B16L: Lab 4 – Android and Shell Scripting

Course Number: 40

Contact Hours per Week: 3

Number of Credits: 3

PART A - ANDROID

This course aims to provide hands on experience on android application development. It also enhances the capability of the students to design and implement solutions to real world computational problems.

- Learners are expected to use laboratory and practice the problems listed in the program list given below.
- Lab cycle consists of 20 experiments.
- Minimum number of experiments in the Laboratory Record Book should be 15.

Course Outcomes:

CO1: To apply the knowledge of android programming for solving the simple real world computing problems

CO2: Create android applications using simple widgets.

CO3: Create android applications using simple adapters and adapter views.

CO4: Create android applications that stores data across sessions

Program List:

1. Write an android application to toast a message on button click.
2. Write an android application to navigate to a new screen on button click.
3. Write an android application to display sum, difference, product and quotient of two numbers
4. Write an android application to generate multiplication table
5. Write an android application to copy the content of an EditText to a TextView.
6. Write an android application to change the background color of an activity
7. Write an android application to change the background image of an activity
8. Create android application with Radio Button control
9. Create android application with Check box control
10. Write android application using Date picker Time picker
11. Write android application to display current date and time using Date picker Dialog Time picker Dialog
12. Write android application using Spinner
13. Write android application using ListView
14. Write android application using GridView
15. Write an android application using Array adapter to populate an adapter view
16. Write an android application to display Alert Dialog
17. Write an android application to display images using ImageView
18. Write an android application to display option menu
19. Create an android application to display a timer using chronometer
20. Design an login page

References:

1. Pro Android 4, SatyaKomatineni & Dave MacLean, Apress.

2. Professional Android 4 Application Development, Retomeier, Wrox.
3. Programming Android, ZigurdMednieks, Laird Dornin, G. Blake Meike, and Masumi Nakamura, O'Reilly

PART B - Shell Scripting

The course aims to offer students a hands on experience on Shell programming concepts using problem oriented learning.

- Learners are expected to use laboratory and practice the problems listed in the program list given below.
- Minimum number of experiments in the Laboratory Record Book should be 15.

Course Outcomes:

CO1: Experiment with basic Shell commands.

CO2: Make use of arithmetic operations in Shell scripting.

CO3: Apply string related techniques in Shell scripting.

Program List:

1. Write a shell script to understand basic shell navigation commands .
2. Write a shell script to print the system date and time.
3. Write a shell script to check whether the given number is zero, positive or negative.
4. Write a shell script to check whether the given number is odd or even.
5. Write a shell script to find the greatest of three numbers.
6. Write a shell script to calculate sum and average
7. Write a shell script to calculate the sum of digits of a number.
8. Write a shell script to find the reverse of a number.
9. Write a shell script to find Fibonacci numbers within a limit.

10. Write a shell script to find the factorial of a number.
11. Write a shell script to find the sum and average of numbers using cmd arguments.
12. Write a shell script to find whether the number is prime or not.
13. Write a shell script to create a simple calculator.
14. Write a shell script to print good morning/good evening.
15. Write a shell script to count number of words, characters.
16. Write a shell script to reverse a string and check whether a given string is palindrome or not.
17. Write a shell script to find the value of one number raised to the power of another. Two numbers are entered through the keyboard.
18. Write a shell script which whenever gets executed displays the message Good Morning/Good afternoon /Good Evening depending on the time it get executed”
19. Write a shell script to find number of vowels, consonants, numbers in a given string.
20. Write a shell script to accept student number, name, marks in 5 subjects. Find total, average and grade.

Rules:

$Avg \geq 90$ then grade A

$Avg < 90$ AND $Avg \geq 80$ then grade B

$Avg < 80$ AND $Avg \geq 60$ then grade C

$Avg < 60$ AND $Avg \geq 40$ then grade D

$Avg < 40$ then grade E

BCS6B17P:Project Work

Course Number: 42

Contact Hours per Week: 2

Number of Credits: 2

Course Description:

This course offer students a glimpse into real world problems and challenges that requires IT based solutions. It provides hand on experience for developing software solution by using software engineering and IT concepts.

Course Outcomes:

CO1: Students will be able to develop a IT based solution for real world Problems.

Course Outline:

The objective of the B.Sc Computer Science final project work is to develop a quality software solution by following the software engineering principles and practices. During the development of the project the students should involve in all the stages of the software development life cycle (SDLC). The main objective of this project course is to provide learners a platform to demonstrate their practical and theoretical skills gained during five semesters of study in B. Sc Computer Science Programme. During project development students are expected to define a project problem, do requirements analysis, systems design, software development, apply testing strategies and do documentation with an overall emphasis on the development of a robust, efficient and reliable software systems. The project development process has to be consistent and should follow standard. For example database tables designed in the system should match with the E-R Diagram. SRS documents to be created as per IEEE standards. Students are encouraged to work on a project preferably on a live software project sponsored by industry or any research organization. Topics selected should be complex and large enough to justify as a B. Sc Computer Science final semester project. The courses studied by the students during the B. Sc Computer Science Programme provide them the comprehensive background knowledge on diverse subject areas in computer science such as computer programming, data

structure, DBMS, Computer Organization, Software Engineering, Computer Networks, etc., which will be helping students in doing project work. Students can also undertake group project to learn how to work in groups. For internal evaluation, the progress of the student shall be systematically assessed through two or three stages of evaluation at periodic intervals. A bonafied project report shall be submitted in hard bound complete in all aspects.



Audit Courses

AUD1E01:Environmental Studies

Course Number: 7

Contact Hours per Week: 0

Theory Number of Credits: 4

Course Description:

This course aims to create an awareness of environment, Ecosystem and Biodiversity Environmental pollution and other social issues

Course Outcomes:

CO1: Students will be able to explain the reasons of environmental pollution and remedies

CO2: Students will be able to summarize the importance of biodiversity preservation

CO3: Students will be able to explain the social issues of pollution

Course Outline:

Module 1 Introduction to Environmental Studies- Multidisciplinary nature of environmental studies, Definition, scope and importance, Need for public awareness, Natural resources and associated problems.

Module 2 Ecosystem- Concept of ecosystems, structure and function of an ecosystem, producers, consumers and decomposers, energy flow in the ecosystem, food chains, food webs and ecological pyramids. Features and functions of forest ecosystem, grassland ecosystem, desert ecosystem and aquatic ecosystems.

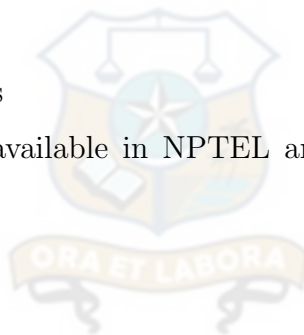
Module 3 Biodiversity and its conservation- Introduction, definition, genetic, species and ecosystem diversity, bio geographical classification of India, value of biodiversity, threats to biodiversity, endangered and endemic species of India, conservation of biodiversity : In-situ and Ex-situ conservation of biodiversity.

Module 4 Environmental Pollution- Definition, cause, effects and control measures of air pollution, water pollution, soil pollution, marine pollution, noise pollution, thermal pollution and nuclear hazards. Solid waste Management: Causes, effects and control measures of urban and industrial wastes, role of an individual in prevention of pollution.

Module 5 Social Issues and the Environment- from Unsustainable to Sustainable development, urban problems related to energy, water conservation, rain water harvesting, watershed management, environmental ethics: Issues and possible solutions. Overview of Environment Protection Act.

References:

1. Online resources
2. Video lectures available in NPTEL and SWAYAM online learning platforms



AUD2E02:Disaster Management

Course Number: 15

Contact Hours per Week: 0

Theory Number of Credits: 4

Course Description:

This course covers various aspects of managing disastrous conditions

Course Outcomes:

CO1: Students will be able to explain different types of disasters

CO2: Students will be able to explain different contexts of disasters

CO3: Students will be able to explain about disasters in India .

Course Outline:

Module 1 Meaning of Disasters, Definitions, Hazard and Disaster, Natural vs. Man-made Disasters, Nature of Disasters

Module 2 Aggravating factors of Disasters, Significance and Repercussions.

Module 3 Understanding various types of disasters, Geological disasters, Climatic disasters, Environmental disasters, Epidemics, Industrial and technological accidents, Causes and effects of disasters

Module 4 Disaster in global context and regional context, Efforts to mitigate disasters worldwide.

Module 5 Disasters in India-Flood, Drought, Cyclone, Earthquake, Landslide.

References:

1. Online resources
2. Video lectures available in NPTEL and SWAYAM online learning platforms.

AUD3E03:Human Rights

Course Number: 22

Contact Hours per Week: 0

Theory Number of Credits: 4

Course Description:

This course aims to create an awareness on human rights, policies and its challenges.

Course Outcomes:

CO1: Students will be able to explain the importance of human rights

CO2: Students will be able to explain about policies and challenges in human rights.

Course Outline:

Module 1 Concept of Human rights, meaning of human rights, definitions, characteristics, evolution of human rights, Adoption of Universal Declaration of Human Rights, Approaches to the study of Human Rights, Classification of Human Rights

Module 2 United Nations and Human Rights: International Bill Of Rights, Universal Declaration of the Human Rights, the UN Commission on Human Rights

Module 3 Indian Constitution and Human Rights: Fundamental Rights, Right To Information Act.

Module 4 Protecting Human Rights: Indian Judiciary and Human Rights, National Human Rights Commission, Role of Media in Protecting Human Rights.

Module 5 Challenges to Human Rights in India, Violence against Women, Police Atrocities against Children, State and Human Rights.

References:

1. Online resources
2. Video lectures available in NPTEL and SWAYAM online learning platforms.

AUD4E04:Gender Studies

Course Number: 30

Contact Hours per Week: 0

Theory Number of Credits: 4

Course Description:

This course provides knowledge about gender issues, laws and policies

Course Outcomes:

CO1: Students will be able to explain gender issues

CO2: Students will be able to explain gender laws and policies

Course Outline:

Module 1 Making women visible- Women in History– Women’s work, Men’s property - Space and Location of Women. Key Concepts and Terminologies- Sex– Gender and Gendering – Sexuality– Patriarchy– Matriarchy –Matriliney – Patriliney–Domestic Violence–House hold Management.

Module 2 Scanning the Indian Society through Women Perspective - Brahmanical patriarchy–Wifhood– Seed and Earth–Sthreedharma – Pathivratha concepts, Widowhood, Food as a medium of proliferation of caste, Three-fold oppression of Dalit women, Caste and Gender.

Module 3 Women in the Fight for Freedom and Social Justice in India- Women in the Revolt of 1857, Participation of Santhal, Bhil and Munda women in anti-British struggles, British attitude towards women fighters. Eminent followers of Mahatma Gandhi, Women in Revolutionary Movements, Women Regiment of INA.

Module 4 Women in the Fight for Freedom and Social Justice in Kerala- Women question in the social reform movement in Kerala, Freedom movement and Kerala women

Module 5 Women in Narratives - Goddess and Dasis in Vedas– Sakuntala- Jatakas and Buddhist literature.

References:

1. Online resources
2. Video lectures available in NPTEL and SWAYAM online learning platforms.



Open Courses

BCS5D01: Office Automation

Course Number: 35

Contact Hours per Week: 3

Theory Number of Credits: 3

Course Description:

This course provides fundamental knowledge office automation, DBMS and web-site development.

Course Outcomes:

CO1: Students will be able to illustrate various functionalities in MS Word and OpenOffice Writer

CO2: Students will be able to illustrate various functionalities in MS Excel and OpenOffice Calc

CO3: Students will be able to illustrate various functionalities in MS PowerPoint and OpenOffice Impress

CO4: Students will be able to illustrate various functionalities in MS Access and OpenOffice Base

CO5: Students will be able to illustrate Website development using Joomla

Course Outline:

Module 1 Documentation Using a Word Processor (OpenOffice Writer / M.S. Word): Introduction to Office Automation, Creating Editing Document, Formatting Document, Auto-text, Autocorrect, Spelling and Grammar Tool, Document Dictionary, Page Formatting, Bookmark. Advance Features - Tables, linking and embedding object, Mail Merge, Macros, Template.

Module 2 Electronic Spread Sheet (OpenOffice Calc/MS-Excel) - Introduction to Spread Sheet, Creating Editing Worksheet, Formatting and Essential Operations, Formulas and Functions, Charts, Advanced features - Pivot table Pivot Chart, Linking and Consolidation

Module 3 Presentation using (OpenOffice Impress/MS-Power Point): Presentations, Creating, Manipulating Enhancing Slides, Organizational Charts, Charts, Word Art, Layering art Objects. Animations and Sounds: Inserting Animated Pictures or Accessing through Object, Inserting Recorded Sound Effect or In-Built Sound Effect

Module 4 Database Management systems using (MS Office-MS Access/ Open Office-Base). Introduction, planning a Database, creating a New Database, Creating Tables, Working with Forms, creating queries, Finding Information in Databases, Creating Reports, Types of Reports, Printing Print Preview

Module 5 Website development using Joomla: Joomla Basics, Installing Joomla and Configuring Joomla. Joomla FrontEnd: Understanding Basic Joomla Template. Working with and Creating Content for Joomla: Defining Content, Managing Content, Managing Categories, Menus and Menu Items, Working with Parameters and Options, Editing Layout

References:

1. Russell A. Stultz, Learn Microsoft Office, BPB Publication.
2. Jennifer Marriott and Elin Waring, The official Joomla Book, Addison Wesley.
3. www.w3schools.com
4. www.html-5-tutorial.com

BCS5D02:Online Marketing

Course Number: 35

Contact Hours per Week: 3

Theory Number of Credits: 3

Course Description:

This course provides fundamental knowledge on eCommerce, online payments and e business models

Course Outcomes:

CO1: Students will be able to explain about eCommerce

CO2: Students will be able to summarize about various e commerce platforms

CO3: Students will be able to explain online payments, e business models and branding

Course Outline:

Module 1 Overview of Internet Marketing: Introduction to E-Marketing, E-Marketing Plan, Practices in E- Marketing, E-Marketing Mix (Product, Pricing, Place), Viral Marketing and Social Networking Marketing, The Rise of Mass Media and Changing Marketing Mindset.

Module 2 Internet Properties & Marketing Implications: Website analysis, Web and other online content, Website properties, Website design and usability features, Multimedia and interactive features

Module 3 Consumer Characteristics Branding Strategy: Branding and consistency online/offline branding, Environmental, situational, and personal factors affecting users, Market segments served, Brand position, Offline versus online customers. The PROS and CONS of online shopping, Justify an Internet business, Internet marketing techniques, The E-cycle of Internet marketing, Personalization e-commerce

Module 4 Business Model, Business pyramid model, Specific e-business models used, Customer value, Revenue stream sources. E-Marketing Metrics and Site objectives. Role of Internet in B2B Application. Planning for Electronic Commerce: Planning Electronic Commerce initiatives, linking objectives to business strategies, measuring cost objectives, Comparing benefits to Costs, Strategies for developing electronic commerce web sites

Module 5 Electronic Data Exchange: Introduction, Concepts of EDI and Limitation, Applications of EDI, Disadvantages of EDI, EDI model. Electronic Payment System: Introduction, Types of Electronic Payment System, Payment Types, Traditional Payment, Value Exchange System, Credit Card System, Electronic Fund Transfer, Paperless bill, Modern Payment Cash, Electronic Cash. E- Governance in India: E- Governance of India, Indian customer EDI System, Service center, Imports, Exports.

References:

1. Parasuraman, A. Zinkhan, G. M. (2003). Marketing to and serving customers through the Internet: An overview and research agenda. *Journal of the Academy of Marketing Science*, 30(4), 286-295
2. G.S.V.Murthy, E-Commerce Concepts, Models, Strategies, Himalaya Publishing House
3. Kamlesh K Bajaj and Debjani Nag, E- Commerce.
4. Gray P. Schneider, Electronic commerce.

BCS5D03:Content Management Systems

Course Number: 35

Contact Hours per Week: 3

Theory Number of Credits: 3

Course Description:

This course provides fundamental knowledge on content management systems.

Course Outcomes:

CO1: Students will be able to explain about content management systems

CO2: Students will be able to summarize about various content management systems like WordPress, Joomla etc.

Course Outline:

Module 1 Content Management System: Introduction, Features, Web-based publishing, format management, revision control, indexing, search, and retrieval. Digital Asset Management System, Distinguish between the basic concepts of user and content, content management application (CMA), Content display application (CDA). Content Management Framework (CMF).

Module 2 Exploring Content Management Systems terminology: Open source, PHP, MySQL, server-side, client-side, static HTML website, how CMS web pages are generated, Website strategy and planning, site mapping, content planning

Module 3 Web content management system: Web Servers, Web hosting, working with WAMP, XAMPP, LAMP- installation and directory structure

Module 4 Website development using Joomla: Joomla Basics, Installing Joomla and Configuring Joomla. Joomla Front End: Understanding Basic Joomla Template. Working with and Creating Content for Joomla: Defining Content, Managing Content, Managing Categories, Menus and Menu Items, Working with Parameters and Options, Editing Layout.

Module 5 Joomla Back End: Joomla Global Configuration, Article Manager, Archive Manager, Frontpage Manager, Section Manager, Category Manager, Media Manager, Menu Manager, Component Manager, Content Manager, Extensions Manager, Module Manager. Practical Application: Basic Planning of Business Sites, Social Media, E-commerce Web Sites.

References:

1. H. M. Dietel, Internet and World Wide Web, Pearson.
2. Jennifer Marriott and Elin Waring, The official Joomla Book, Addison Wesley.
3. www.w3schools.com



Electives

BCS6E101: Cloud Computing

Course Number: 41

Contact Hours per Week: 4

Theory Number of Credits: 3

Course Description:

The course is prepared to introduce the cloud computing concepts and map reduce programming model. The course provides skills and knowledge about operations and management in cloud technologies. It enables the learner to understand the fundamentals of cloud infrastructure.

Course Outcomes:

- CO1:** Explain the evolution, principles, and benefits of Cloud Computing in order to assess existing cloud infrastructures.
- CO2:** Decide a suitable model to capture the user needs by interpreting different service delivery and deployment models.
- CO3:** Outline virtualization foundations to cater the needs of elasticity, portability and resilience by cloud service providers.
- CO4:** Infer Open source architectural style and identify platforms like EUCALYPTUS, Open Nebula and Open Stack.
- CO5:** Outline the cloud applications using map reduce programming models and Hadoop architecture.

Course Outline:

Module 1 Introduction - evolution of cloud computing – system models for distributed and cloud computing – NIST cloud computing reference architecture – Infrastructure as a Service (IaaS) – resource virtualization – Platform as a Service (PaaS) – cloud platform & management – Software as a Service (SaaS) – available service providers.

Module 2 Virtualization - basics of virtualization - types of virtualization – implementation levels of virtualization - virtualization structures - tools and mechanisms - virtualization of CPU, memory, I/O devices - desktop virtualization – server virtualization – Linux KVM, Xen, Qemu, LXC, OpenVZ.

Module 3 Programming model - parallel and distributed programming paradigms – Mapreduce, twister and iterative Mapreduce – mapping applications - programming support – Apache Hadoop – HDFS, Hadoop I/O, Hadoop configuration, MapReduce on Hadoop.

Module 4 Programming model - parallel and distributed programming paradigms – Mapreduce, twister and iterative Mapreduce – mapping applications - programming support – Apache Hadoop – HDFS, Hadoop I/O, Hadoop configuration, MapReduce on Hadoop.

Module 5 Security in the cloud - security overview – cloud security challenges – software as a service security – security governance – risk management – security monitoring – security architecture design – data security – application security – virtual machine security – Qubes – desktop security through Virtualization.

References:

1. Kai Hwang, Geoffrey C Fox, Jack G Dongarra, Distributed and Cloud Computing (From Parallel Processing to the Internet of Things), Elsevier Science, ISBN: 9780128002049.
2. John W. Rittinghouse and James F. Ransome, Cloud Computing: Implementation, Management, and Security, 1st Edition, CRC Press, ISBN: 1439806802.
3. Toby Velte, Robert Elsenpeter and Anthony Velte, Cloud Computing, A Practical Approach, TMH, ISBN: 9780071626958.
4. George Reese, Cloud Application Architectures, 1st Edition, Shroff /O'Reilly, ISBN: 8184047142.

5. Ravi Nair and Jim Smith, Virtual Machines: Versatile Platforms for Systems and Processes, 1st Edition, Elsevier Science / Morgan Kaufmann, ISBN: 9780080525402/1558609105.
6. Katarina Stanoevska - Slabeva, Thomas Wozniak, Santi Ristol, Grid and Cloud Computing – A Business Perspective on Technology and Applications, Springer, ISBN: 3642051928.



BCS6E102:Visual Programming

Course Number: 41

Contact Hours per Week: 4

Theory Number of Credits: 3

Course Description:

The course enables the learners to understand the basic architecture of Visual programming platforms. It covers the .Net architecture and programming using VB.Net

Course Outcomes:

- CO1:** Students will be able to illustrate the architecture of .Net framework
- CO2:** Students will be able to use the control structures in Vb.Net
- CO3:** Students will be able to create form based applications using Vb.net
- CO4:** Students will be able to summarize the object orient concepts in .Net framework
- CO5:** Students will be able to illustrate database connectivity using ADO.Net

Course Outline:

Module 1 Introduction to visual programming - Concept of event driven programming - Introduction to VB.Net environment, the .NET Framework and the Common Language Runtime. Building VB.NET Applications, the Visual Basic Integrated Development - Basic Language – Console application and windows application, Data types, Declaring Variables, scope of variables, operators and statements.

Module 2 Making Decisions with If . . . Else Statements, Using Select Case, Making Selections with Switch and Choose, Loop statements - Do Loop, for, while – The With Statement – Handling Dates and Times - Converting between Data Types - Arrays - declaration and manipulation - Strings string functions – Sub Procedures and Functions.

Module 3 Windows Applications - Forms - Adding Controls to Forms, Handling Events, MsgBox, InputBox, Working with Multiple Forms, Setting the Startup Form, SDI MDI Forms, Handling Mouse Keyboard Events, Common controls (Text Boxes, Rich Text Boxes, Labels, Buttons, Checkboxes, Radio Buttons, Group Boxes, List Boxes, Checked List Boxes, Combo Boxes, Picture Boxes, Scroll Bars, Tool Tips, Timers) properties – methods.

Module 4 Object-Oriented Programming - Creating and using Classes objects – Handling Exceptions - On Error GoTo - Raising an Exception - Throwing an Exception - Using Structured Exception Handling - Debugging and tracing.

Module 5 Data Access with ADO.NET - Accessing Data with the Server Explorer - Accessing Data with Data Adaptors and Datasets - Creating a New Data Connection - Creating and populating Dataset - Displaying Data in a Data Grid - Selecting a Data Provider - Data Access Using Data Adapter Controls – Binding Data to Controls - Handling Databases in Code - Binding to XML data.

References:

1. Visual Basic .NET Black Book, by Steven Holzner
2. VB.NET for developers, By Keith Franklin, Rebecca Riordan, SAMS
3. Sams Teach Yourself Visual Studio .NET 2005 in 21 Days, By Jason Beres.
4. Learning Visual Basic .NET by Jesse Liberty.
5. Visual Basic .Net programming in easy steps BY TIM ANDERSON, DreamTech Press.

BCS6E103:Micro Processor & Micro Controllers

Course Number: 41

Contact Hours per Week: 4

Theory Number of Credits: 3

Course Description:

The course enables the learners to understand the basic architecture of microprocessors and micro controllers. This course focuses on the architecture, assembly language programming, interrupts, interfacing of microprocessors with peripheral devices and the basic concepts of microcontrollers. It helps the learners to extend the study of latest processors and understand hardware-based solutions.

Course Outcomes:

- CO1:** Illustrate the architecture, modes of operation and addressing modes of microprocessors.
- CO2:** Develop 8086 assembly language programs.
- CO3:** Demonstrate interrupts, its handling and programming in 8086.
- CO4:** Illustrate how different peripherals (8255,8254,8257) and memory are interfaced with microprocessors.
- CO5:** Outline features of microcontrollers and develop low level programs

Course Outline:

Module 1 8085 microprocessor (-Basic Architecture only). 8086 microprocessor – Architecture and signals, Physical Memory organization, Minimum and maximum mode of 8086 system and timings. Comparison of 8086 and 8088. Machine language Instruction format.

Module 2 Addressing Modes of 8086. Instruction set – data copy /transfer instructions, arithmetic instructions, logical instructions, string manipulation instructions, branch instructions, unconditional and conditional branch instruction, flag manipulation and processor control instructions. Assembler Directives and operators. Assembly Language Programming with 8086.

Module 3 Stack structure of 8086, programming using stack- Interrupts - Types of Interrupts and Interrupt Service Routine- Handling Interrupts in 8086- Interrupt programming. Programmable Interrupt Controller - 8259, Architecture (Just mention the control word, no need to memorize the control word)- Interfacing Memory with 8086.

Module 4 Programmable Peripheral Input/output port 8255 - Architecture and modes of operation- Programmable interval timer 8254-Architecture and modes of operation- DMA controller 8257 Architecture (Just mention the control word, no need to memorize the control word of 8254 and 8257)

Module 5 8051 Architecture- Register Organization- Memory and I/O addressing- Interrupts and Stack- 8051 Addressing Modes- Instruction Set- data transfer instructions, arithmetic instructions, logical instructions, Boolean instructions, control transfer instructions- Simple programs.

References:

1. Barry.B.Brey. The Intel Microprocessor 8086/8088. 80186, 80286, 80386 and 80486 Architecture Programming and Interfacing, Prentice Hall of India Pvt.Ltd.1995.
2. K.R. Venugopal, Raj Kumar, Microprocessor X86 programming, Bpb publications New Delhi
3. Abel P, IBM PC Assembly Language Programming 5th Edition Pearson Education Asia 2001
4. B Ram, Fundamentals of Microprocessors and Microcomputers – Dhanpat Rai Publications Pvt. Ltd., New Delhi
5. Mohamad Rafiqzaman, Microprocessors and Microcomputer Based System Designing. Universal Bookstall, New Delhi
6. Yu. Cheng Liu, Glenn A Gibson, Microcomputer Systems: The 8086/8088 Family. Architecture, Programming Designing Prentice Hall of India.
7. A. NagoorKani, Microprocessors and Microcontrollers, Second Edition, Tata McGraw Hill
8. Ramesh Gaonkar, Microprocessor Architecture, Programming, and Applications with the 8085, Penram International Publishing Pvt. Ltd.

BCS6E104:Introduction to Internet of Things-IOT

Course Number: 41

Contact Hours per Week: 4

Theory Number of Credits: 3

Course Description:

The course enables the learners to understand the basic concepts of Internet of Things, its architecture, Arduino and Raspberry –Pi

Course Outcomes:

- CO1:** Illustrate the architecture and uses of IoT
- CO2:** Explain about the Arduino environment
- CO3:** Summarize the working of Arduino based system.
- CO4:** Illustrate the use of Raspberry –Pi
- CO5:** Outline features of Python for Pi

Course Outline:

Module 1 Introduction to Arduino environment: Overview, board description, installation, program structure, data types, variables constants, operators, control statements, loops, functions, strings, string objects, time, arrays.

Module 2 Arduino Function libraries: I/O functions, Advanced I/O functions, Character functions, math library, trigonometric functions. Arduino Advanced: Due bZero, Pulse width Modulation, Random Numbers, Interrupts, Communication, Inter Integrated circuit, Serial peripheral Interface

Module 3 Introduction to Raspberry –Pi, SOC, versions, features, Raspberry Pi B+ , Basics of Electronics: Hardware Description, Pin Configuration, Preparing SD Card for OS Installation: Downloading Image, Various Operating Systems Available, Making SD Card: Fedora ARM Installer Raspberry Pi SD Installer: First time configuration: Using Whole SD Card, Using Whole Screen, Changing Timezone, Booting Into Desktop, Raspi-Config Test.

Module 4 Network Setup: Setting Up Using GUI, Setting up using command line, Finding Pi's IP Address. GPIO Setup: The GPIO Connector, Libraries Using GIT, Configuring GPIO: Test and Configure. Pi using Secure Shell (SSH): Enabling SSH, Putty, Test and Configure. Pi over VNC: Installing VNC, Configuring VNC, Running VNC at Startup, Test and Configure

Module 5 Python for Pi: Importing Libraries, Functions, Classes, Accessing SMTP Inbox Using Python, Manipulating GPIO Pins Using Python. Making Raspberry –Pi a Web Server: Apache installation, Configuring and Deploying Apache, Hosting a website.

References:

1. Arduino Cookbook: Recipes to Begin, Expand, and Enhance Your Projects, Michael Margolis, O' Reilly
2. Raspberry Pi- Beginners Guide, Gareth Halfacree, Raspberry Press
3. Online resources

