

DAYANANDA SAGAR UNIVERSITY
SHAVIGE MALLESHWARA HILLS, KUMARASWAMY LAYOUT
BENGALURU-560 111, KARNATAKA.

SCHOOL OF ENGINEERING



SCHEME & SYLLABUS
FOR
BACHELOR OF COMPUTER APPLICATIONS
(BCA)

(1st & 2nd Semesters)

(With effect from 2021-22)

SEMESTER/YEAR : **I SEM / I YEAR**
COURSE CODE : **21CA1101**
TITLE OF THE COURSE : **MATHEMATICS-I**
L: T: P: S/P: C : **03-01-00-00-04**

Course Objectives The Curriculum supports the prerequisites to enhance their Mathematical knowledge towards their understanding mathematical Concepts and help them to persuade research work in concerned fields with the help of Mathematical approach.

Course outcomes At the end of the course student will be able to
1. understand and use counting principle, use Propositional calculus in Theorem proving
2. use suitable algebraic structures to model the given scenario/system
3. use constructions used in proofs as algorithms

Module 1 **MATRICES** 09 hrs
Definition, Types of Matrices, Addition, Subtraction, Scalar Multiplication and Multiplication of Matrices, Adjoint, Inverse, Eigen values and Eigen Vectors of a Matrix, Caley-Hamilton Theorem (Statement only) Rank of a matrix, Row reduced echelon form and normal form

Solution of homogeneous and nonhomogeneous system of equations.
Practical's: Solving problems under matrices using FOSS TOOLS SCILAB/Maxima
DETERMINANTS:

Module 2 **SETS AND RELATIONS** 09 hrs
Definition, Minors, Cofactors, Properties of Determinants Cramer's rule
SETS:
Sets, Subsets, Types of Sets, Operation on Sets, Applications.
RELATIONS AND FUNCTIONS:
Definition, Types of functions, Types of relations with illustrations and graphs

Module 3 **DIFFERENTIAL CALCULUS** 09 hrs
Limit , Continuity, Differentiation , Product rule Quotient rule Successive differentiation, Leibnitz theorem for finding nth derivative of product of functions(only statement)
Partial derivatives, homogeneous functions Euler's theorem (only statement) maxima and minima,
Taylor's series and Maclaurin's Series (without proof).

Practical's: Solutions Using FOSS TOOLS SCILAB/MAXIMA

Module 4 **DIFFERENTIAL EQUATIONS** 08 hrs
Different types of differential equations, solving different equations

Module 5 **ALGEBRAIC STRUCTURES** 10 hrs
Introduction, operations, semi-groups, groups, subgroups, normal subgroups, Isomorphism

and homomorphism, rings, integral domains and fields, Polynomials over a field.

Textbooks

1. Dr. P. R. Vittal : MATHEMATICAL FOUNDATION

Reference
Books

2. Shanti Narayan, "Differential Calculus", S Chand & Company.
3. Shanti Narayan, "Matrices and determinant", S. Chand and company LTD
4. B.S. Grewal, "Elementary Engineering Mathematics", 34th Ed. Delhi Khanna Publishers.
5. Das BC and Mukherjee, Differential Calculus, Calcutta, U.N. Dhar Publishers.

I SEM / I YEAR COURSE CODE : 21CA1102
TITLE OF THE COURSE : FUNDAMENTALS OF PROGRAMMING
L: T: P: S/P : C : 03-00-00-03

Course Objectives The objective of the subject is to teach the student the basics of Programming starting with simple problem solving techniques implemented using Python language

Course outcomes At the end the course the students will

1. understand Programming concepts, building blocks of a program, different constructs file i/o operations
2. Be able to design using advanced concepts such as functions, modules and regular expressions in designing programs.

Module 1	INTRODUCTION TO PROGRAMMING Introduction to problem solving, algorithms, flowcharts, art of Programming language, programming environments, overview of compilations. Data types, constants and variables, Arithmetic expressions, operators. Arithmetic expressions, evaluation of arithmetic expression, type casting and conversion, operator hierarchy & associativity	09 hrs
Module 2	BUILDING BLOCKS Branching: Decision making with IF statement, IF-ELSE statement, Nested IF statement, ELSE-IF ladder, switch statement, goto statement. break, continue Statement	09 hrs
Module 3	Iterative statements: For, while, and do-while loop, jumps in loops. Arrays, pointers and structures.	08 hrs
Module 4	SEQUENCES , FILES AND DICTIONARIES Sequences, FILE I/O Operations, dictionaries and sets	09 hrs
Module 5	PROGRAMMING CONCEPTS Functions , modules and regular expressions	10 hrs

- Text Books**
1. Programming in Python 3 (Second Edition) A Complete Introduction to the Python Language by Mark Summerfield
 2. Head First Python – Paul Barry
 3. Gottfried, Byron S., Programming with C, Tata McGraw Hill

ReferenceBooks

1. Problem Solving With C: Jones and Harrow, WILEY INDIA PVT LTD.
2. An Introduction to Programming and Problem Solving With Pascal, G. Michael Schneider Steven W. Weingart David M. Perlman, WILEY INDIA PVT LTD.

SEMESTER/YEAR : **I SEM / I YEAR**
COURSE CODE : **21CA1103**
TITLE OF THE COURSE : **COMPUTER ORGANIZATION - I**
L: T: P: S/P : C : **03-00-00-00-03**

Course Objectives : This course aims at providing an overview of basic digital design techniques, basic structure of a digital computer and its organization.

Course outcomes : At the end of the course student will be able

1. to demonstrate knowledge of binary number theory, Boolean algebra and binary codes.
2. to analyse and design combinational systems using
3. standard gates and minimization methods (such as Karnaugh maps,
4. Quine-McCluskey Algorithm).

 To analyse and design combinational systems composed of Standard combinational modules, such as multiplexers and decoders.

5. to analyse the given problems and design sequential Circuits

Module 1	INTRODUCTION TO DIGITAL COMPUTER Concept of Digital Computer, Types of Software – System software / Application software / Utility Software. Compilers, Interpreters, Assemblers, Linker, Loader. System Bus Structure	06 hrs
Module 2	NUMBER SYSTEM AND LOGIC DESIGN MINIMIZATION TECHNIQUES Introduction to number system, Signed Binary Representation, Logic gates and Boolean Algebra.	06 hrs
Module 3	COMBINATIONAL AND SEQUENTIAL LOGIC CIRCUITS Minimization Techniques and circuit design, Codes and Arithmetic operation, Combinational Circuits Design	06 hrs
Module 4	COMBINATIONAL AND SEQUENTIAL LOGIC CIRCUITS(CONTD) Multiplexers, De-multiplexers, decoders, Sequential circuit Design Counter and registers	06 hrs
Module 5	INPUT/ OUTPUT AND MEMORY ORGANIZATION OF COMPUTER I/O Devices, Memory Systems, Interrupts and instruction cycle, Addressing modes	06 hrs

Text Book:

1. M. Morris Mano, “Digital Logic and Computer Design”, 10th Edition, Pearson Education, 2004.
2. Malvino. D. Leach, “Digital Principles and Applications”, 5th Edition, Tata McGraw Hill.
3. Floyd, “Digital Principles”, Pearson Education, 2002

Reference Books

1. Carl Hamacher, Z Varnesic and S. Zaky: Computer Organization, 5th Edition, McGraw Hill, 2002.
2. W. Stalling, “Computer Organization and Architecture”, 8th Edition, Pearson Education India, 2010.

SEMESTER/YEAR : **I SEM / I YEAR**
COURSE CODE : **21CA1105**
TITLE OF THE COURSE : **ACCOUNTING AND FINANCIAL MANAGEMENT**
L: T: P: S/P: C : **03-00-02-00-04**

Course Objectives This paper provides conceptual knowledge of Financial accounting, familiarizes with methods of preparing Final Accounts of Sole Proprietorship concerns, provides understanding of the accounting procedure for different kinds of businesses and various ratios and abstracts of accounts which are required for the management to take strategic decisions.

Course outcomes At the end of the course student will be able
 1. To understand the principles and methodology of Nature of Financial Accounting
 2. To know Basics of accounting
 3. To do Analysis of financial statement
 4. To know Indian economy. Fund flow Cash flow.

Module 1 Overview – Meaning and Nature of Financial Accounting, Scope of Financial Accounting, Financial Accounting and Management Accounting, Accounting concepts and conventions, Accounting standards. **06 Hour**

Module 2 Basics of accounting – Capital & Revenue items, Introduction to Double Entry System, Introduction to preparation of Journal, Ledger, Procedure for Recording and Posting journal entries and ledgers, Introduction to Trail Balance, Preparation of Final Account, Profit & Loss Account and related concepts, Balance Sheet and related concept. **06 Hour**

Module 3 Analysis of financial statement: Introduction to Ratio Analysis, Uses of Ratio analysis, Simple problems of Ratio analysis, Analysis of Balance Sheet, Break-even analysis: Uses of Break – even analysis, simple problem solving in break-even analysis. **09 Hour**

Module 4 Fund flow and Cash Flow: Preparation of Fund Flow statement, Analysis of Fund Flow statement, Schedule of Changes in Working Capital, **08 Hour**

Module 5 Preparation of Cash Flow Statement, Analysis of Fund Flow statement. **09 Hour**

Text Books 1) Narayanswami, “Financial Accounting: A Managerial Perspective”, PHI, 2nd Edition.
 2) Mukherjee, “Financial Accounting for Management”, TMH, 1st Edition.

Reference Books 1) Gupta Ambrish, “Financial Accounting for Management”, Pearson Education, 2nd Edition. (2) Chowdhary, Anil, “Fundamentals of Accounting and Financial Analysis”, Pearson Education, 1st Edition.

Software: Tally – Not an open source, Wings (Open Source) , Dynamics GP. (Not open source).

SEMESTER/YEAR : **I SEM / I YEAR**
COURSE CODE : **21CA1106**
TITLE OF THE COURSE : **ENGLISH IN PRACTICE**
L: T: P: S/P: C : **02-00-02-00-03**

Course Objectives

- a. To enable students improve their lexical, grammatical competence.
- b. To enhance their communicative skills.
- c. To equip students with oral and appropriate written communication skills.
- d. To inculcate students with employability and job search skills.

Course outcomes

1. Students achieve proficiency in English
2. Develop their professional communication skills
3. Acquire skills for placement

Module 1 **Grammar and Vocabulary:** Tense and Concord, word formation, Homonyms and Homophones 06 hrs

Module 2 **Listening and Speaking:** Common errors in Pronunciation (Individual sounds); Process description (Describing the working of a machine, and the manufacturing process), use of vocabulary and rendering. 06 hrs

Module 3 **Group Discussion** 06 hrs

Module 4 **Writing:** Interpretation of data (Flow chart, Bar chart), Referencing Skills for Academic Report Writing 06 hrs

Module 5 **Reading:** Reading Comprehension, Answering questions, Appreciation of creative writing. 06 hrs

Text Books	1.	Dhanavel.S.P. English and Communication Skills for Students of Science and Engineering, Orient Blackswan Ltd., 2009.
	2.	Meenakshi Raman and Sangeetha Sharma. Technical Communication- Principles and Practice, Oxford University Press, 2009
Reference Books	1.	Day.R A., Scientific English:A Guide for Scientists and Other Professional, 2nd ed. Hyderabad: Universities Press, 2000

SEMESTER/YEAR : **II SEM / I YEAR**
COURSE CODE : **21CA1202**
TITLE OF THE COURSE : **OBJECT ORIENTED PROGRAMMING USING JAVA**
L: T: P: S/P: C : **03-00-00-00-03**

Course Objectives This subject covers the basics of OOPS concepts such as Inheritance, Polymorphism, Event driven programming, Exceptions and Concurrent programming.
Course outcomes The students will
 i) understand OOP concepts, programming blocks. Packages and interfaces.
 ii) understand and use the exception handling and design programs with multithreading.

Module 1	Basics of Object Oriented Programming (OOP) Need for OO paradigm, Classes and objects, Constructors and Destructors, Access control, Inheritance, Abstraction, Encapsulation, Polymorphism, Overriding, String handling.	9 Hour
Module 2	Programming blocks Data types , variables, Identifiers, Key words, scope and life time of variables, operators, expressions, control statements, type conversion and casting.	9 Hour
Module 3	Packages and Interfaces Defining, Creating and Accessing a Package, importing packages, Interfaces, abstract Classes, implementing interface, variables in interface, extending interfaces.	9 Hour
Module 4	Exception handling Concepts of exception handling, benefits of exception handling, Termination or presumptive models, exception hierarchy, built in exceptions, creating own exception sub classes.	9 Hour
Module 5	Multithreading : Thread life cycle, creating threads, synchronizing threads, daemon threads, and thread groups.	09 Hour

- Text Books**
 1. Java: How to Program, 8/e, Dietal, Dietal, PHI
 2. The C++ Programming Language , Bjarne Stroustrup
 3. Java: The complete reference, 7/e, Herbert Schildt, TMH.

SEMESTER/YEAR : **II SEM / I YEAR**
COURSE CODE : **21CA1203**
TITLE OF THE COURSE : **DATA STRUCTURES**
L: T: P: S/P: C : **03-00-00-00-03**

Course objectives The objective of the subject is to teach the student the usage of data structures using any Programming language.

Course outcomes At the end of the course student will be able
1. To understand and explain linear and dynamic allocation of memory using array and linked list
2. To design programs for queue, stacks, sorting, Searching using linear data structures.
3. To use Nonlinear data structures to create Tree, Graphs.

Module 1	Searching Techniques Preliminaries of algorithm, Algorithm analysis and complexity. Recursion: Definition, Design Methodology and Implementation of recursive algorithms, Linear and binary recursion, recursive algorithms for factorial function, GCD computation, Fibonacci sequence, Towers of Hanoi, Tail recursion List Searches using Linear Search, Binary Search, Fibonacci Search.	10 Hour
Module 2	Sorting Techniques Basic concepts, Sorting by : insertion (Insertion sort), selection (heap sort), exchange (bubble sort, quick sort), distribution (radix sort) and merging (merge sort) Algorithms.	10 Hour
Module 3	Stacks and Queues Basic Stack Operations, Representation of a Stack using Arrays, Stack Applications: Reversing list, Factorial Calculation, In-fix- to postfix Transformation, Evaluating Arithmetic Expressions. Queues: Basic Queues Operations, Representation of a Queue using array, Implementation of Queue Operations using Stack, Applications of Queues-Round robin Algorithm, Enqueue, Dequeue, Circular Queues, Priority Queues.	10 Hour
Module 4	Linked Lists Introduction, single linked list, representation of a linked list in memory, Operations on a single linked list, merging two single linked lists into one list, Reversing a single linked list, applications of single linked list to represent polynomial expressions and sparse matrix manipulation, Advantages and disadvantages of single linked list, Circular linked list, Double linked list	10 Hour
Module 5	Trees and advance Data Structures Basic tree concepts, Binary Trees: Properties, Representation of Binary Trees using arrays and linked lists, operations on a Binary tree , Binary Tree Traversals (recursive), Creation of binary tree from in-order and pre (post) order traversals. Adv Data Structures	10 Hour
Text Books	1. Data Structures, 2/e, Richard F, Gilberg , Forouzan, Cengage 2. Data Structures and Algorithms, 2008,G.A.V.Pai, TMH	
Reference Books	1. Data Structure with C, Seymour Lipschutz, TMH 2. Fundamentals of Data Structure in C, 2/e, Horowitz, Sahni, Anderson Freed, University Press	

SEMESTER/YEAR : **II SEM / I YEAR**
COURSE CODE : **21CA1204**
TITLE OF THE COURSE : **Computer Organization II**
L: T: P: S/P: C : **03-00-00-00-03**

Course Objectives : This course aims to equip the student with assembly level programming using 8086 microprocessor.

Course outcomes : At the end of the course student will be able
 1. to understand the design the data part and control part of a processor
 2. to understand memory technology, I/O systems and I/O operation and use them in the design of a computing system
 3. To understand and appreciate the 8086 processor and program the same to solve problems.

Module 1 **Introduction to the microprocessor and computer:** 10 hrs
 Internal processor architecture, Functional block diagram, Bus, Clock signals, addressing modes.

Module 2 **Programming 8086:** 10 hrs
 Data movement instructions, Arithmetic and logic instructions, Program control instructions, string instructions, programming techniques, examples.
 Modular programming: stacks, subroutines, Macros.

Module 3 **Assembler Directives:** 10 hrs
 Data Definition and Storage Allocation, Program Organization, Alignment, Program End, Value Returning Attribute, Procedure Definition, Macro Definition, Data Control, Branch Displacement, Header File, Inclusion, Target Machine Code, Generation Control Directives.

Module 4 Interrupts and interrupt routines, I/O interface and programming: Fundamental I/O considerations, Data transfer schemes, Programmed I/O, Interrupt I/O, DMA, System bus structure, Min.Max Modes, Application of 8259, 8255, 8251, 8257, 8253

Module 5 **Introduction to other 16 bit 32 bit processors:** 80286, 386, 486, Pentium and Pentium Pro-processor. 10 hrs

Text Books
 1. YuChang Liu & Glenn A Gibson, "Microcomputer systems: the 8086\8088 Family: Architecture, Programming and design", PHI
 2. Microprocessor 8086 : Architecture, Programming and Interfacing : Mathur Sunil, PHI Learning

Reference Books
 1. Douglas V. Hall- Microprocessors and digital systems, MH.
 2. Kenneth L. Short - Microprocessor and Programmed Logic ", PHI, 2nd Edition.

SEMESTER/YEAR : **II SEM / I YEAR**
COURSE CODE : **21CA1205**
TITLE OF THE COURSE : **TECHNICAL COMMUNICATIONS**
L: T: P: S/P: C : **02-00-02-00-03**

Course Objectives To teach the elements of effective writing and communicative methods

Course outcomes 1. The student will be able to communicate effectively orally and in written
2. Draft technical reports and proceedings.

Module 1	Preparation of Abstract, Synopsis Notices	06 hrs
Module 2	Technical Paper writing, Minutes of the meeting	06 hrs
Module 3	Letter Writing(Letters of enquiry, Permission, Regret, Reconciliation, Complaint, Breaking the ice.)	06 hrs
Module 4	Drafting Curriculum Vitae, Resume and Covering Letters. Job Applications	06 hrs
Module 5	Memo, E-mail Etiquette.	06 hrs

Text Books 1. N. Krishnaswamy and T. Sri Raman, Creative English for communication, Macmillan Publication-2005.
2. Meenakshi Raman & Sangeeta Sharma, Technical Communication – Principles and Practice, oxford University press

Reference Books 1. N. Krishnaswamy and T. Sriraman, Creative English for Communication Business Communication and Report Writing, Macmillan.

DAYANANDA SAGAR UNIVERSITY
SHAVIGE MALLESHWARA HILLS, KUMARASWAMY LAYOUT
BENGALURU – 560 111, KARNATAKA.

SCHOOL OF ENGINEERING



SCHEME & SYLLABUS FOR MASTER OF COMPUTER APPLICATIONS

(I, II, III & IV Semester)

SEMESTER/YEAR : **I SEM / I YEAR**
COURSE CODE : **21MCA4101**
TITLE OF THE COURSE : **Data Structures**
L: T/A: P: C : **3 : 0 : 0 : 3**

COURSE OBJECTIVES:

- To study the basics and advanced concepts of C programming language.
- To learn the concepts of linear data structures and its applications.
- To understand the concepts of non-linear data structures.
- To learn the usage of sorting techniques.
- To familiarize the concepts of hashing.

COURSE OUTCOMES:

On completion of the course, the students will be able to:

1. Demonstrate basic and advanced concepts of C programming language.
2. Use abstract data types including stacks, queues and lists for any application.
3. Design and implement tree data structures.
4. Analyze and implement hashing techniques that solve in linear time.
5. Apply sorting algorithms for a given problem.
6. Choose appropriate data structure and implement a given application.

UNIT 1 - LINEAR DATA STRUCTURES **9**

Abstract Data Types (ADTs) – List ADT – Array-Based Implementation – Linked List Implementation – Doubly-Linked Lists – Circular Linked Lists – Stack ADT: Implementation of Stacks – Queue ADT: Implementation of Queues – Applications.

UNIT 2 - HIERARCHICAL **9**

Trees: Preliminaries – Implementation of Trees – Tree Traversals with an Application – Binary Trees: Implementation – Expression Trees – Search Tree ADT – Binary Search Trees – Applications of Trees.

UNIT 3 - HEAP STRUCTURES **10**

Binary Heaps – Min Max Heaps – Leftist Heaps – Skew Heaps – Binomial Heaps – Fibonacci Heaps – Lazy Merging for Binomial Queues – Fibonacci Heap Operations – Amortized Analysis.

UNIT 4 - HASHING AND SORTING

12

Fundamentals of Hashing – Hash Function – Separate Chaining – Open Addressing – Linear Probing – Quadratic Probing – Double Hashing – Rehashing – Extendible Hashing -Sorting Algorithms: Insertion Sort, Shell Sort, Quick Sort, Heap Sort, Merge Sort.

UNIT 5 - GRAPH ALGORITHMS

9

Graphs: Representation of Graphs – Graph Traversals – Topological sort – Shortest Path Algorithms: Dijkstra's Algorithm – Graph with Negative Edge Costs – All Pairs Shortest Path -Minimum Spanning Tree: Prim's and Kruskal's Algorithm.

References:

1. Brian W. Kernighan, Dennis Ritchie, "The C Programming Language", Second Edition, Pearson Education, 2015.
2. Brian W. Kernighan, Rob Pike, "The Practice of Programming", Pearson Education, 1999.
3. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C", Second Edition, Pearson Education, 1997.
4. Y. Langsam, M. J. Augenstein, A. M. Tenenbaum, "Data Structures using C", Pearson Education Asia, 2004.
5. V. Alfred, J. E. Hopcroft, J. D. Ullman, "Data Structures and Algorithms", Pearson education Asia, 1983.
6. Robert Kruse & Bruce Leung, "Data Structures & Program Design in C", Pearson Education, 2007.

SEMESTER/YEAR : **I SEM / I YEAR**
COURSE CODE : **21MCA4102**
TITLE OF THE COURSE : **Computer Architecture and Microcontroller**
L: T/A: P: C : **3 : 0 : 2 : 4**

COURSE OBJECTIVES:

1. This course introduces the principles of the basic computer architecture and microcontroller concepts.
2. The course emphasizes performance and cost analysis, instruction set design, pipelining, memory technology, memory hierarchy, virtual memory management, and I/O systems.

COURSE OUTCOMES:

At the end of the course students will be able:

1. To understand the principles of the basic computer architecture and microcontroller concepts.
2. Design Memory Interfacing circuits.
3. Design and implement 8051 microcontroller based systems.

MODULE 1 - PIPELINING

Basic Concepts – Data Hazards – Instruction Hazards – Influence On Instruction Sets – DataPath And Control Considerations – Performance Considerations – Exception Handling.

MODULE 2 - MEMORY SYSTEM

Basic Concepts – Semiconductor Ram – Rom – Speed – Size And Cost – Cache Memories – Improving Cache Performance – Virtual Memory – Memory Management Requirements – Associative Memories – Secondary Storage Devices.

MODULE 3 - I/O INTERFACING

Memory Interfacing and I/O interfacing - Parallel communication interface – Serial communication interface – D/A and A/D Interface - Timer – Keyboard /display controller – Interrupt controller – DMA controller – Programming and applications Case studies: TrafficLight control, LED display , LCD display, Keyboard display interface and Alarm Controller.

MODULE 4 - MICROCONTROLLER

Architecture of 8051 – Special Function Registers(SFRs) - I/O Pins Ports and Circuits - Instruction set - Addressing modes - Assembly language programming.

MODULE 5 - INTERFACING MICROCONTROLLER

Programming 8051 Timers - Serial Port Programming - Interrupts Programming – LCD & Keyboard Interfacing - ADC, DAC & Sensor Interfacing - External Memory Interface- Stepper Motor and Waveform generation.

Text Books:

1. Carl Hamacher, Zvonko Vranesic and Safwat Zaky, “Computer Organization”, Fifth Edition, Tata Mcgraw Hill, 2002.
2. Mohamed Ali Mazidi, Janice Gillispie Mazidi, Rolin McKinlay, “The 8051 Microcontroller and Embedded Systems: Using Assembly and C”, Second Edition, Pearson Education, 2011

References Books:

1. David A. Patterson and John L. Hennessy, “Computer Organization And Design: TheHardware/Software Interface”, Third Edition, Elsevier, 2005.
2. William Stallings, “Computer Organization And Architecture – Designing For Performance”, Sixth Edition, Pearson Education, 2003.
3. Douglas V.Hall, “Microprocessors and Interfacing, Programming and Hardware:, TMH, 2012.

SEMESTER/YEAR : **I SEM / I YEAR**
COURSE CODE : **21MCA4103**
TITLE OF THE COURSE : **Advanced Operating Systems**
L: T/A: P: C : **3 : 0 : 0 : 3**

COURSE OBJECTIVES:

1. To understand the advanced Operating Systems, their design and implementation.
2. To understand the implementation in UNIX/LINUX, Windows and Embedded OS.

COURSE OUTCOMES:

At the end of the course students will be able to:

1. Skill of designing OSs
2. Implementations of Design strategies in existing OSs and Distributed Process Management.

MODULE 1 - INTRODUCTION TO OS AND SHELL SCRIPTING

Operating system overview-objectives and functions, Evolution of Operating System.- Computer System Organization- Operating System Structure and Operations- System Calls, System Programs, OS Generation and System Boot.

Basics of Shell scripting, writing scripting program, command line arguments, Functions and file manipulators, regular expressions and filters

MODULE 2 - USING THE OPERATING SYSTEMS & OPERATING SYSTEMS ORGANIZATION

The programmer's abstract machine; Resources; Processes and threads; Writing concurrent programs. Basic functions; General implementation considerations; Contemporary OS kernels.

MODULE 3 - DESIGN STRATEGIES

Design considerations; Device Drivers, Monolithic kernels; Modular organization; Microkernel; Layered organizations; Operating Systems for distributed system.

MODULE 4 - DISTRIBUTED SYSTEMS

Networking; The Need for a Protocol Architecture; The TCP/IP Sockets; Linux Networking; Client/Server Computing; Distributed Message Passing; Remote Procedure Calls; Clusters; Windows Vista Cluster Server; Linux Clusters.

MODULE 5 - RTOS (ANDROID)

Basic introduction to Android and Case studies

Text Books:

1. Gary Nutt: Operating Systems, 3rd Edition, Pearson, 2004.
2. William Stallings: Operating Systems: Internals and Design Principles, 6th Edition, Prentice Hall, 2008.
3. John Horton: Android Programming for beginners, 1st Edition, 2015

Reference Books:

1. Silberschatz, Galvin, Gagne: Operating System Concepts, 8th Edition, Wiley, 2008
2. Andrew S. Tanenbaum, Albert S. Woodhull: Operating Systems, Design and Implementation, 3rd Edition, Prentice Hall, 2006.
3. Pradeep K Sinha: Distribute Operating Systems, Concept and Design, PHI, 2007.

SEMESTER/YEAR : **I SEM / I YEAR**
COURSE CODE : **21MCA4104**
TITLE OF THE COURSE : **Advanced Software Engineering and Testing**
L: T/A: P: C : **3 : 0 : 2 : 4**

COURSE OBJECTIVES:

1. Understand agile software development practices
2. Demonstrate Agile development and testing techniques
3. Know the benefits and pitfalls of working in an Agile team
4. Understand agile development and testing.

COURSE OUTCOMES:

At the end of the course students will be able to:

1. At the end the student will be able to
2. The know importance of interacting with business stakeholders in determining the requirements for a software system.
3. Apply iterative software development process
4. Apply the impact of social aspects on software development success.

MODULE 1 - AGILE METHODOLOGY

9

Theories for Agile management – agile software development – traditional model vs. agile model - classification of agile methods – agile manifesto and principles – agile project management – agile team interactions – ethics in agile teams - agility in design, testing – agile documentations – agile drivers, capabilities and values.

MODULE 2 - AGILE PROCESSES

9

Lean production - SCRUM, Crystal, Feature Driven Development, Adaptive Software Development, and Extreme Programming: Method overview – lifecycle – work products, roles and practices.

MODULE 3 - AGILITY AND KNOWLEDGE MANAGEMENT

9

Agile information systems – agile decision making – Earl’s schools of KM – institutional knowledge evolution cycle – development, acquisition, refinement, distribution, deployment , leveraging – KM in software engineering – managing software knowledge – challenges of migrating to agile methodologies – agile knowledge sharing – role of story-cards – Story-card Maturity Model (SMM).

MODULE 4 - AGILITY AND REQUIREMENTS ENGINEERING

9

Impact of agile processes in RE – current agile practices – variance – overview of RE using agile – managing unstable requirements – requirements elicitation – agile requirements abstraction model – requirements management in agile environment, agile requirements prioritization – agile requirements modeling and generation – concurrency in agile requirements generation.

MODULE 5 - SOFTWARE TESTING

9

Testing as an Engineering Activity – Testing Fundamentals – Defects – Strategies and Methods for Black Box Test Case Design – Strategies and Methods for White-Box Test Case design – Test Adequacy Criteria – Evaluating Test Adequacy Criteria – Levels of Testing and different types of testing – OO Testing- Agile testing- methodologies

Text Books:

1. Dingsoyr, Torgeir, Dyba, Tore, Moe, Nils Brede (Eds.), –Agile Software Development, Current Research and Future Directions||, Springer-Verlag Berlin Heidelberg, 2010
2. David J. Anderson; Eli Schragenheim, –Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results||, Prentice Hall, 2003
3. Hazza& Dubinsky, –Agile Software Engineering, Series: Undergraduate Topics in Computer Science||, Springer, VIII edition, 2009
4. Craig Larman, –Agile and Iterative Development: A manager_s Guide||, Addison-Wesley, 2004
5. Kevin C. Desouza, –Agile information systems: conceptualization, construction, and management, Butterworth-Heinemann, 2007.
6. M G Limaye, “Software Testing – Principles, Techniques and Tools”, McGraw Hill, 2011.

SEMESTER/YEAR : **I SEM / I YEAR**
COURSE CODE : **21MCA4105**
TITLE OF THE COURSE : **Probability & Statistics**
L: T/A: P: C : **3 : 0 : 2 : 4**

COURSE OBJECTIVES:

This course aims at providing the required skill to apply the statistical tools in engineering problems.

COURSE OUTCOMES:

At the end of the course students will be able to:

1. The students will have a fundamental knowledge of the concepts of probability.
2. Have knowledge of standard distributions which can describe real life phenomenon.
3. Have the notion of sampling distributions and statistical techniques used in engineering and management problems.

MODULE 1 - RANDOM VARIABLES **9**

Discrete and continuous random variables – Moments – Moment generating functions – Binomial, Poisson, Geometric, Uniform, Exponential, Gamma and Normal distributions.

MODULE 2 - TWO – DIMENSIONAL RANDOM VARIABLES **9**

Joint distributions – Marginal and conditional distributions – Covariance – Correlation and Linear regression – Transformation of random variables – Central limit theorem (for independent and identically distributed random variables).

MODULE 3 - TESTING OF HYPOTHESIS **9**

Sampling distributions – Estimation of parameters – Statistical hypothesis – Large sample test based on Normal distribution for single mean and difference of means -Tests based on t,Chisquare and F distributions for mean, variance and proportion – Contingency table (test for independent) – Goodness of fit.

MODULE 4 - DESIGN OF EXPERIMENTS **9**

Impact of agile One way and Two way classifications – Completely randomized design – Randomized block design – Latin square design – 22 factorial design.

Classification – Stationary process – Markov process – Poisson process – Discrete parameter Markov chain – Chapman Kolmogorov equations – Limiting distributions.

Control charts for measurements (X and R charts) – Control charts for attributes (p, c and np charts) – Tolerance limits – Acceptance sampling.

Text Books:

1. Milton. J. S. and Arnold. J.C., “Introduction to Probability and Statistics”, Tata McGraw Hill, 4th Edition, 2007.
2. Johnson. R.A. and Gupta. C.B., “Miller and Freund’s Probability and Statistics for Engineers”, Pearson Education, Asia, 7th Edition, 2007.
3. Papoulis. A and Unnikrishnapillai. S., “Probability, Random Variables and Stochastic Processes” McGraw Hill Education India, 4th Edition, New Delhi, 2010.

Reference Books:

1. Devore. J.L., “Probability and Statistics for Engineering and the Sciences”, Cengage Learning, New Delhi, 8th Edition, 2012.
2. Walpole. R.E., Myers. R.H., Myers. S.L. and Ye. K., “Probability and Statistics for Engineers and Scientists”, Pearson Education, Asia, 8th Edition, 2007.
3. Ross, S.M., “Introduction to Probability and Statistics for Engineers and Scientists”, 3rd Edition, Elsevier, 2004.
4. Spiegel. M.R., Schiller. J. and Srinivasan. R.A., “Schaum’s Outline of Theory and Problems of Probability and Statistics”, Tata McGraw Hill Edition, 2004.

SEMESTER/YEAR : **I SEM / I YEAR**
COURSE CODE : **21MCA4106**
TITLE OF THE COURSE : **Data Structures Lab**
L: T/A: P: C : **0 : 0 : 4 : 2**

COURSE OBJECTIVES:

1. Be familiarized with good programming design methods, particularly Top- Down design.
2. Getting exposure in implementing the different data structures using C Appreciate recursive algorithms.

COURSE OUTCOMES:

At the end of the course students will be able to:

1. Design and implement C programs for manipulating stacks, queues, linked lists, trees, and graphs.
2. Apply good programming design methods for program development.
3. Apply the different data structures for implementing solutions to practical problems.
4. Develop recursive programs using trees and graphs.

Lab experiments and mini projects will include C programming concepts and implementation of ADT data structures using C.

SEMESTER/YEAR : I SEM / I YEAR
COURSE CODE : 21MCA4107
TITLE OF THE COURSE : **Advanced Operating Systems Lab**
L: T/A: P: C : **0 : 0 : 4 : 2**

COURSE OBJECTIVES:

1. To have practical implementation of OS
2. To implement design strategies for Operating System.
3. To design Distributed systems and give implementation aspects of distributed Process management.

COURSE OUTCOMES:

At the end of the course students will be able to:

1. Understanding the design and implementation aspects OS
2. Understand and develop skills to design distributed systems and process Management

Lab experiments and mini projects will include Shell scripting, concurrent programming, process management, socket programming, client server computing, Parallel programming, distributed message passing, remote procedure calls, and distributed process management.

SEMESTER/YEAR : **II SEM / I YEAR**
COURSE CODE : **21MCA4201**
TITLE OF THE COURSE : **Advanced Algorithms**
L: T/A: P: C : **3 : 0 : 0 : 3**

COURSE OBJECTIVES:

1. To understand the design of advanced algorithms and data structures.
2. To understand the applications of algorithms in different fields such as geometry, number theory, signal processing and linear algebra.

COURSE OUTCOMES:

At the end of the course students will be able to:

1. Skill of advanced algorithm design.
2. Knowledge of advanced data structures.

MODULE 1 - MULTI-THREADED ALGORITHMS AND MATRIX OPERATIONS

The basics of dynamic multi-threading; multi-threaded versions of matrix multiplication, solution of linear equations, matrix inversion, and least squared approximation.

MODULE 2 - FAST FOURIER TRANSFORMS AND NUMBER THEORETIC ALGORITHMS

Representation of polynomials, DFT and FFT; efficient FFT implementation (sequential and parallel). Elementary number theoretic notions, greatest common divisor, modular arithmetic, solving modular linear equations, Chinese remainder theorem, powers of an element, RSA public crypto system, primality testing, integer factorization.

MODULE 3 - STRING MATCHING ALGORITHMS

Rabin-Karp, Knuth-Morris-Pratt and Boyer-Moore string matching algorithms. Suffix trees and their applications in computational biology.

MODULE 4 - COMPUTATIONAL GEOMETRY AND PROBABILISTIC ANALYSIS

Line segment properties, determining whether pair of line segments intersects, finding the convex hull, finding the closest pair of points. Hiring problem, indicator random variables, randomized algorithms; probabilistic analysis and further uses of indicator random variables.

MODULE 5 - ADVANCED DATA STRUCTURES AND GRAPH ALGORITHMS

Fibonacci heaps, mergeable heap operations, decreasing a key and deleting a node, bounding the maximum degree. Van emde boas tree. Data structures for disjoint sets, analysis of union by rank with path compression. Flow networks, Ford-Fulkerson method, maximum bipartite matching.

Text Books:

Thomas H. Cormen, Charles E Leiserson, Ronald L. Rivest, and Clifford Stein, Introduction to Algorithms., 3rd ed., The MIT Press, 2009.

Reference Books:

Donald E Knuth, Art of Computer Programming, Volumes 1-4A, Addison-Wesley, 2011.

SEMESTER/YEAR : **II SEM / I YEAR**
COURSE CODE : **21MCA4203**
TITLE OF THE COURSE : **COMPUTER COMMUNICATION AND NETWORKS**
L: T/A: P: C : **3: 0 : 0 : 3**

COURSE OBJECTIVES:

1. To understand the network fundamentals.
2. To explore various application layer protocols.
3. To understand the transport layer services.
4. To learn the network layer functionalities.
5. To understand the link layer services and data communication fundamentals.

COURSE OUTCOMES:

At the end of the course students will be able to:

1. Describe the fundamentals of internetworking.
2. Design new application layer protocols for various applications.
3. Select suitable transport layer protocols for network applications.
4. Trace and analyze the packets between end-to-end applications.
5. Calculate the capacity of links between nodes.
6. Identify suitable signal encoding techniques for various scenarios.

MODULE 1 - APPLICATION LAYER

9 Hrs

Principles of Network Applications – Protocol Architecture: TCP/IP Protocol Architecture – Introduction to Socket Programming – Web and HTTP – File Transfer Protocol – Domain Name System – SMTP – POP – IMAP – Peer to Peer Applications – Performance: Latency, Delay and Bandwidth Product.

MODULE 2 - TRANSPORT LAYER

9 Hrs

Transport Layer Services – UDP – Reliable Byte Stream: TCP, Connection Establishment and Termination, Sliding Window, Triggering Transmission, Adaptive Retransmission, TCP Extensions – TCP Congestion Control.

MODULE 3 - NETWORK LAYER

9 Hrs

Network Layer Functions – Switching and Forwarding: Datagrams, Virtual Circuit Switching, Source Routing – Internetworking: IPv4, IP Address Classes, Subnetting, CIDR, ARP, DHCP, ICMP – Routing: DVR, LSR, BGP – IPv6.

MODULE 4 - DATA LINK LAYER

9 Hrs

Hardware Building Blocks: Nodes, Links – Link Layer Functions – Framing: PPP, HDLC – Error Detection: Two Dimension Parity, Checksum, CRC – Reliable Transmission: Stop and Wait, Sliding Window – MAC: Ethernet, WiFi, Bluetooth.

MODULE 5 - FUNDAMENTALS OF DATA COMMUNICATION

9 Hrs

Communication Model – Data communications – Data Transmission: Concepts and Terminology, Analog and Digital Transmission, Transmission Impairments – Signal Encoding Techniques: Digital Data and Digital Signals – Multiplexing: FDM, TDM, Multiple Channel Access.

Text Books:

1. James F. Kurose, Keith W. Ross, “Computer Networking: A Top-Down Approach”, Seventh Edition, Pearson Education, 2017.
2. Larry L. Peterson and Bruce S. Davie, “Computer Networks: A Systems Approach”, Fifth Edition, Morgan Kaufmann Publishers, 2011.

References:

1. William Stallings, “Data and Computer Communications”, Tenth Edition, Pearson, 2014.
2. Ying-Dar Lin, Ren-Hung Hwang, Fred Baker, “Computer Networks: An Open Source Approach”, McGraw-Hill, 2012.

SEMESTER/YEAR : **II SEM / I YEAR**
COURSE CODE : **21MCA4204**
TITLE OF THE COURSE : **DATABASE MANAGEMENT SYSTEMS**
L: T/A: P: C : **3 : 0 : 0 : 3**

COURSE OBJECTIVES:

- To learn the fundamentals of data models, conceptualize and depict a database system using ER diagram.
- To study the principles to be followed to create an effective relational database and write SQL queries to store/retrieve data to/from database systems.
- To know the fundamental concepts of transaction processing, concurrency control techniques and recovery procedure.
- To learn about the internal storage structures using different file and indexing techniques and the basics of query processing and optimization.
- To study the basics of distributed databases, semi-structured and un-structured data models.

COURSE OUTCOMES:

1. Model an application's data requirements using conceptual modeling and design database schemas based on the conceptual model.
2. Formulate solutions to a broad range of query problems using relational algebra/SQL.
3. Demonstrate an understanding of normalization theory and apply such knowledge to the normalization of a database.
4. Run transactions and estimate the procedures for controlling the consequences of concurrent data access.
5. Explain basic database storage structures, access techniques and query processing.
6. Describe distributed, semi-structured and unstructured database systems.

UNIT 1 - RELATIONAL DATABASES

9

Purpose of Database System – Views of Data – Data Models – Database System Architecture
– Introduction to Relational Databases – Relational Model – Keys – Relational Algebra – Relational Calculus – SQL Fundamentals – Advanced SQL features – Triggers – Embedded SQL.

UNIT 2 - DATABASE DESIGN **9**

Entity-Relationship Model – ER Diagrams – Functional Dependencies – Non-Loss Decomposition
Functional Dependencies – First Normal Form – Second Normal Form – Third Normal Form –
Dependency Preservation – Boyce/Codd Normal Form – Multi-Valued Dependencies and Fourth
Normal Form – Join Dependencies and Fifth Normal Form.

UNIT 3 - TRANSACTION MANAGEMENT **9**

Transaction Concepts – ACID Properties – Serializability – Transaction Isolation Levels –
Concurrency Control – Need for Concurrency – Lock-Based Protocols – Deadlock Handling
– Recovery System – Failure Classification – Recovery Algorithm.

UNIT 4 - IMPLEMENTATION TECHNIQUES **9**

Overview of Physical Storage Media – RAID – File Organization – Organization of Records in Files –
Indexing and Hashing – Ordered Indices – B+ tree Index Files – Static Hashing – Dynamic Hashing –
Query Processing Overview – Catalog Information for Cost Estimation – Query Optimization.

UNIT 5 - ADVANCED TOPICS **9**

Overview of Distributed Databases – Data Fragmentation – Replication – XML Databases – XML Schema
– NOSQL Database: Characteristics – CAP theorem – Types of NoSQL Databases: Column Oriented,
Document, Key-Value and Graph Types – Applications – Current Trends.

Text Books:

1. Abraham Silberschatz, Henry F. Korth, S. Sudharshan, “Database System Concepts”, Sixth Edition, Tata McGraw Hill, 2014.
2. Ramez Elmasri, Shamkant B. Navathe, “Fundamentals of Database Systems”, Seventh Edition, Pearson Education, 2017.

References:

1. C. J. Date, A. Kannan, S. Swamynathan, “An Introduction to Database Systems”, Eighth Edition, Pearson Education, 2006.
2. Raghuram Ramakrishnan, Johannes Gehrke, “Database Management Systems”, Fourth Edition, Tata McGraw Hill, 2010.
3. G. K. Gupta, “Database Management Systems”, Tata McGraw Hill, 2011.
4. Carlos Coronel, Steven Morris, Peter Rob, “Database Systems: Design, Implementation and Management”, Ninth Edition, Cengage Learning, 2011.

SEMESTER/YEAR : **II SEM / I YEAR**
COURSE CODE : **21MCA4205**
TITLE OF THE COURSE : **Advanced Algorithm Lab**
L: T/A: P: C : **0 : 0 : 4 : 2**

COURSE OBJECTIVES:

1. To understand the design and implementation of advanced algorithms and data structures.
2. To implement the applications of algorithms in different fields such as geometry, number theory, signal processing and linear algebra.

COURSE OUTCOMES:

At the end of the course students will be able to:

1. Be skilled in advanced algorithm design and implementation.
2. Knowledge of advanced data structures.

LIST OF EXPERIMENTS:

Implementation in the following topics:

Lab experiments and mini projects will include dynamic multi-threaded versions of matrix multiplication, solutions of linear equations, Fast Fourier Transforms – DFT and FFT, advanced string matching algorithms, computational geometry, probabilistic analysis, advanced data structures and graph algorithms.

SEMESTER/YEAR : **II SEM / I YEAR**
COURSE CODE : **21MCA4206**
TITLE OF THE COURSE : **JAVA PROGRAMMING AND NETWORKS LABORATORY**
L: T/A: P: C : **0 : 0 : 4 : 2**

COURSE OBJECTIVES:

1. To understand and apply the fundamentals of core Java.
2. To implement inheritance, polymorphism, interfaces, multithreading, streaming, networking, generic collections and RMI.
3. To develop web applications using client side and server side programming.
4. To understand SOAP and REST based web service standards.
5. To learn and use MVC architecture for application development.

COURSE OUTCOMES:

At the end of the course students will be able to:

1. Write programs on advanced features of Java such as streaming, networking, multithreading and generics.
2. Design and develop GUI based components and animations.
3. Develop chat and file transfer applications.
4. Create JDBC based distributed applications using RMI and Java Beans.
5. Develop dynamic data driven websites using server side programming.
6. Create MVC applications using advanced frameworks.

Lab experiments and mini projects will include Java Programming concepts, Java socket programming, JDBC, Remote Method Access using RMI Implementation Database Connectivity using Java Bean, Session Management and Implementation of Cookies using JSF, Web application development using Struts framework & Spring framework and Analyze live HTTP/DNS/UDP/TCP/IP/ICMP/DHCP packets using Wireshark tool.

Mini projects on Real time Applications.

SEMESTER/YEAR : **II SEM / I YEAR**
COURSE CODE : **21MCA4207**
TITLE OF THE COURSE : **DATABASE SYSTEM MANAGEMENT LAB**
L: T/A: P: C : **0 : 0 : 4 : 2**

COURSE OBJECTIVES:

1. To learn and implement important commands in SQL.
2. To learn the usage of nested and joint queries.
3. To understand functions, procedures and procedural extensions of databases.
4. To understand design and implementation of typical database applications.
5. To be familiar with the use of a front end tool for GUI based application development.

COURSE OUTCOMES:

At the end of the course students will be able to:

1. Create databases with different types of key constraints.
2. Write simple and complex SQL queries using DML and DCL commands.
3. Realize database design using 3NF and BCNF.
4. Use advanced features such as stored procedures and triggers and incorporate in GUI based application development.
5. Create XML database and validate with meta-data (XML schema).
6. Create and manipulate data using NOSQL database.

Lab Experiments and Mini Projects will include:

1. Create a database table, add constraints (primary key, unique, check, Not null), insert rows, update and delete rows using SQL DDL and DML commands.
2. Create set of tables, add foreign key constraints and incorporate referential integrity.
3. Query the database tables using different 'where' clause conditions and also implement aggregate functions.
4. Query the database tables and explore sub queries and simple join operations.
5. Query the database tables and explore natural, equi and outer joins.
6. Write user defined functions and stored procedures in SQL.
7. Execute complex transactions and realize DCL and TCL commands.
8. Write SQL Triggers for insert, delete, and update operations in database table.
9. Create View and index for database tables with large number of records.
10. Create a XML database and validate it using XML schema.
11. Create Document, column and graph based data using NOSQL database tools.
12. Develop a simple GUI based database application and incorporate all the above-mentioned feature

Course: Computer Networks

School	School of Engineering
Department	Computer Applications
Programme	B.C.A
Course Title	Computer Networks
Course Code	20CA2301
Year	2020-2023
Semester	III
Name of the Faculty Members	
e-mail id	

1. Course Size and Instruction Method

Credits	L	T	P	Hours/Week
4	3	0	2	3

Total Semester hours: 3 x13= 39 Hours Mentoring Hours: As per time table

2 Course Details

2.1 Course Aim and Summary

The course '**Computer Networks**' aims at introducing networks theory. The course deals with data communication, data link, transport and application layer. The course emphasises on improving networking and routing system. During the course, the students will be taught network layers and its functions.

2.2 Course Objectives

The objectives of the Course are:

- Understand the division of network functionalities into layers.
- Be familiar with the components required to build different types of networks
- Be exposed to the required functionality at each layer
- Learn the flow control and congestion control algorithms

2.3 Course Outcomes

After undergoing this course, students will be able to:

	Outcomes	Bloom's Taxonomy Level
C01	Identify the components required to build different types of networks	1- Remember
C02	Choose the required functionality at each layer for given application	2- Understand
C03	Identify solution for each functionality at each layer	6- Create
C04	Trace the flow of information from one node to another node in the network	3-Apply
C05	Outline concepts of Routing	2- Understand

2.4 Course Content INTRODUCTION:

Building a network – network edge and core – layering and protocols - - OSI Reference Model - Network Topologies – Internet Architecture - networking devices – modems, routers, switches, gateways. **Data Communication:** Signal characteristics – Data transmission – Physical links and transmission media – Signal encoding techniques - Channel access techniques – TDM – FDM.

DATA LINK LAYER AND LAN:

Link layer services – Framing - Error control – flow control – media access control - Ethernet – CSMA/CD – Token Ring - FDDI - Wireless LANs – CSMA/CA.

NETWORK AND ROUTING:

Circuit switching – packet switching – virtual circuit switching - Routing - IP – Global Address – Datagram Forwarding – Subnetting – CIDR - ARP – DHCP – RIP – OSPF - BGP - ICMP – IPv6.

TRANSPORT LAYER:

Overview of Transport layer – UDP - TCP – Reliable byte stream – connection management – flow control – retransmission - Congestion control - congestion avoidance.

APPLICATION LAYER:

Needs/Principles of Application layer Protocols – Web and HTTP – FTP – Electronic Mail (SMTP, POP3, IMAP, MIME) – DNS – SNMP

2.5 Text Book and References Text Book:

1. Larry L. Peterson and Bruce S. Davie, "Computer Networks: A systems approach", Morgan Kaufmann Publishers, USA, 2010.
2. James F. Kurose, Keith W. Ross, "Computer Networking – A Top-Down Approach Featuring the Internet", Pearson Education, New Delhi, 2009.

References:

1. Prakash C Gupta, "Data Communication and Computer Networks", Prentice Hall of India, New Delhi, 2009.
2. Achyut S Godbole, "Data Communication and Networking", Tata McGraw Hill Publishing Company, New Delhi, 2007.
3. Ying-Dar Lin, Ren-Hung Hwang and Fred Baker, "Computer Networks An Open Source Approach", McGraw Hill Publisher, USA, 2011.
4. Andrew S Tanenbaum, David J. Wetherall "Computer Networks", Prentice Hall of India/ Pearson Education, New Delhi, 2010.

Course: Human Computer Interface

School	School of Engineering
Department	Computer Applications
Programme	B.C.A
Course Title	Human Computer Interface
Course Code	20CA2302
Year	2020-2023
Semester	III
Name of the Faculty Members	
e-mail id	

1. Course Size and Instruction Method

Credits	L	T	P	Hours/Week
4	3	0	2	3

Total Semester hours: 3 x13= 39 Hours Mentoring Hours: As per time table

2. Course Details

2.1 Course Aim and Summary

The course '**HUMAN COMPUTER INTERFACE**' aims at introducing. The course deals with design and software process of HCI. The course emphasises on improving mobile application information. During the course, the students will be taught about design and tools of mobile and software process.

2.2 Course Objectives

The objectives of the Course are:

- Learn the foundations of Human Computer Interface
- Be familiar with the design technologies for individuals and persons with disabilities
- Be aware of mobile HCI
- Learn the guidelines for user interface.

2.3 Course Outcomes

After undergoing this course, students will be able to:

	Outcomes	Bloom's Taxonomy Level
C01	Design effective dialog for HCI	1- Remember
C02	Design effective HCI for individuals and persons with disabilities.	2-Understand
C03	Assess the importance of user feedback.	6- Create
C04	Explain the HCI implications for designing multimedia/ e-commerce/ e-learning Web sites	3-Apply
C05	Develop meaningful user interface.	2-Understand

2.4 Course Content FOUNDATIONS OF HCI:

The Human: I/O channels – Memory – Reasoning and problem solving; The computer: Devices – Memory – processing and networks; Interaction: Models – frameworks – Ergonomics – styles – elements – interactivity- Paradigms.

DESIGN & SOFTWARE PROCESS:

Interactive Design basics – process – scenarios – navigation – screen design – Iteration and prototyping. HCI in software process – software life cycle – usability engineering – Prototyping in practice – design rationale. Design rules – principles, standards, guidelines, rules. Evaluation Techniques – Universal Design.

MODELS AND THEORIES:

Cognitive models – Socio-Organizational issues and stake holder requirements – Communication and collaboration models- Hypertext, Multimedia and [WWW](http://www).

MOBILE HCI:

Mobile Ecosystem: Platforms, Application frameworks- Types of Mobile Applications: Widgets, Applications, Games- Mobile Information Architecture, Mobile 2.0, Mobile Design: Elements of Mobile Design, Tools.

WEB INTERFACE DESIGN:

Designing Web Interfaces – Drag & Drop, Direct Selection, Contextual Tools, Overlays, Inlays and Virtual Pages, Process Flow. Case Studies.

2.5 Text Book and References Text Book:

1. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, "Human Computer Interaction", 3rd Edition, Pearson Education, 2004 (UNIT I, II & III)
2. Brian Fling, "Mobile Design and Development", First Edition, O'Reilly Media Inc., 2009 (UNIT -IV)
3. Bill Scott and Theresa Neil, "Designing Web Interfaces", First Edition, O'Reilly, 2009.(UNIT-V)

Course: Software Engineering

School	School of Engineering
Department	Computer Applications
Programme	B.C.A
Course Title	Software Engineering
Course Code	20CA2303
Year	2020-2023
Semester	III
Name of the Faculty Members	
e-mail id	

1. Course Size and Instruction Method

Credits	L	T	P	Hours/Week
3	3	2	0	3

Total Semester hours: 3 x13= 39 **Hours Mentoring Hours:** As per time table

2. Course Details

2.1 Course Aim and Summary

The course '**Software Engineering**' aims at introducing the different phases of software project. The course deals with fundamental concept of requirement and analysis modelling. The course emphasises on improving the software management.

2.2 Course Objectives

The objectives of the Course are:

- Understand the phases in a software project
- Understand fundamental concepts of requirements engineering and Analysis Modelling.
- Understand the major considerations for enterprise integration and deployment.
- Learn various testing and maintenance measures

2.3 Course Outcomes

After undergoing this course, students will be able to:

	Outcomes	Bloom's Taxonomy Level
--	----------	------------------------

C O 1	Identify the key activities in managing a software project.	1- Rememb er
C O 2	Compare different process models	2- Understa nd
C O 3	Concepts of requirements engineering and Analysis Modeling	6- Create
C O 4	Apply systematic procedure for software design and deployment.	3-Apply
C O 5	Concepts of configure management and maintenance	2- Understa nd

2.4 Course Content

SOFTWARE PROCESS

Definition of terms - The evolving role of Software – Software characteristics - Software applications - Software Myths.

Software Process: Software process models - The linear sequential model - The prototyping model - The RAD model - Evolutionary software process models - The incremental model - The spiral model - Fourth generation techniques- Agile process.

SYSTEM ENGINEERING:

Requirements elicitation - Requirements analysis and negotiation - Requirements validation - Requirements management – Joint Application Development – case study.

ANALYSIS MODELING:

The elements of the analysis model - Data modeling - Data objects, attributes and relationships - Cardinality and modality - Entity/Relationship diagram - Data flow diagrams - The data dictionary - Other classical analysis methods – case study.

DESIGN CONCEPTS AND PRINCIPLES:

Modularity - Functional Independence - Cohesion-Coupling - Design documentation. Software architecture - Architectural design - Transform centered architecture - Transaction centered architecture – case study

SOFTWARE METRICS:

Scope – Classification of metrics – Measuring Process and Product attributes – Direct and

2.5 Text Book and References Text Book:

1. Roger S Pressman, Software Engineering: A Practitioner's Approach, McGraw Hill International edition, Seventh edition, 2009.
2. Ian Sommerville, Software Engineering, 8th Edition, Pearson Education, 2008.

References:

1. Stephan Schach, Software Engineering, Tata McGraw Hill, 2007
2. Pfleeger and Lawrence Software Engineering: Theory and Practice, Pearson Education, second edition, 2001

Course: Numerical Methods

School	School of Engineering
Department	Computer Applications
Programme	B.C.A
Course Title	Numerical Methods
Course Code	20CA2304
Year	2020-2023
Semester	III
Name of the Faculty Members	
e-mail id	

1. Course Size and Instruction Method

Credits	L	T	P	Hours/Week
4	3	0	0	3

Total Semester hours: $3 \times 13 = 39$ **Hours Mentoring Hours:** As per time table

2. Course Details

2.1 Course Aim and Summary

The course '**Numerical Methods**' aims at introducing numerical methods and mathematical modelling on problem solving.

2.2 Course Objectives

The objectives of the Course are:

- The Curriculum supports the prerequisites to enhance their Mathematical Knowledge.
- To understand mathematical and numerical concepts.
- To persuade research work in concerned fields with the help of mathematical approach

2.3 Course Outcomes

After undergoing this course, students will be able to:

	Outcomes	Bloom's Taxonomy Level
C01	Understand and use Eigen values, interpolation and approximation in problem solving	1- Remember
C02	Use suitable numerical differentiation and integration model for the given scenario/system	2- Understand
C03	To acquaint the knowledge of various techniques and methods of solving numerical techniques	6- Create
C04	To acquaint the knowledge of various techniques and methods of solving ordinary differential	3-Apply
C05	To understand the knowledge of various techniques and methods of solving various types of partial differential equations	2- Understand

2.4 Course Content

Mathematical Modeling, Numerical Methods and Problem Solving:

A Simple Mathematical Model- Conservation Laws in Engineering and Science- MATLAB Fundamentals. Roots and Optimization-Roots: Bracketing Methods - Open Methods

Linear Systems:

Gauss Elimination - *LU* Factorization- Matrix Inverse and Condition- Iterative Methods.

Curve Fitting:

Linear Regression- General Linear Least-Squares and Nonlinear Regression- Fourier Analysis.

Numerical Differentiation And Integration:

Numerical Integration Formulas - Numerical Integration of Functions- Numerical Differentiation.

Ordinary Differential Equations:

Initial-Value Problems - Adaptive Methods and Stiff Systems - Boundary-Value Problems.

2.5 Text Book and References Text Book:

1. Gerald, C.F, and Wheatley, P.O, "Applied Numerical Analysis", Sixth Edition, Pearson Education Asia, New Delhi, 2002.
2. Kandasamy, P., Thilagavathy, K. and Gunavathy, K., "Numerical Methods", S.Chand Co. Ltd., New Delhi, 2003
3. Balagurusamy, E., "Numerical Methods", Tata McGraw-Hill Pub.Co.Ltd, New Delhi, 1999.
4. Burden, R.L and Faires, T.D., "Numerical Analysis", Seventh Edition, Thomson Asia Pvt. Ltd., Singapore, 2002

Course: Analysis And Design Of Algorithms

School	School of Engineering
Department	Computer Applications
Programme	B.C.A
Course Title	Analysis And Design Of Algorithms
Course Code	20CA2305
Year	2020-2023
Semester	III
Name of the Faculty Members	
e-mail id	

1. Course Size and Instruction Method

Credits	L	T	P	Hours/Week
3	3	0	0	3

Total Semester hours: 3 x13= 39 Hours **Mentoring Hours:** As per time table

2. Course Details

2.1 Course Aim and Summary

The course '**Analysis And Design Of Algorithms**' aims at introducing algorithm analysis techniques. The course deals with different algorithm design and its limitations.

2.2 Course Objectives

The objectives of the Course are:

- Learn the algorithm analysis techniques.
- Become familiar with the different algorithm design techniques.
- Understand the limitations of Algorithm power.

2.3 Course Outcomes

After undergoing this course, students will be able to:

	Outcomes	Bloom's Taxonomy Level
C01	Understand the various methods for analysis of algorithms.	1- Remember
C02	Design algorithms for various computing problems	2- Understand
C03	Analyse the time and space complexity of algorithms	6- Create

C O 4	Critically analyse the different algorithm design techniques for a given problem	3-Apply
C O 5	Modify existing algorithms to improve efficiency	2-Understand

2.4 Course Content INTRODUCTION:

Definition and properties of algorithms- Recurrence and Non Recurrence algorithm - Analysis of algorithms- Asymptotic notations- Solving recurrence relations- Complexity analysis of Insertion sort, Radix sort, Linear search, finding factorial, binary search - Introduction to NP-Hard and NP-Completeness.

DIVIDE-AND-CONQUER AND GREEDY METHOD:

The general method- Finding Maximum and Minimum Element-Quick sort – Merge sort- Matrix multiplication

Greedy Method: The general method- Optimal storage on tapes- Knapsack problem- Minimum spanning trees- Single source shortest path method..

DYNAMIC PROGRAMMING:

The General method- All pairs shortest path- Optimal binary Search tree- Multistage graphs.

BACKTRACKING:

The General method- Solution space and tree organization- The Eight Queens problem - Sum of subset problem - Graph coloring - Knapsack problem.

BRANCH AND BOUND:

The General method- LC search – LC branch and Bound – FIFO branch and bound- 0/1 Knapsack problem- Traveling sales person problem- Efficiency consideration.

2.5 Text Book and References Text Book:

1. Thomas H.Cormen, Charles E. Leiserson, Ronald L Rivest, Clifford Stein, "Introduction to Algorithms", MIT Press, England, 2009.

References:

1. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", Galgotia Publications, New Delhi, 2010.
2. Anany V Levitin, "Introduction to the Design & Analysis of Algorithms", Prentice Hall of India /PearsonEducation,NewDelhi,2008.

Course: Operating System

School	School of Engineering
Department	Computer Applications
Programme	B.C.A
Course Title	Operating System
Course Code	20CA2306
Year	2020-2023
Semester	III
Name of the Faculty Members	
e-mail id	

1. Course Size and Instruction Method

Credits	L	T	P	Hours/Week
3	3	0	0	3

Total Semester hours: 3 x13= 39 Hours Mentoring Hours: As per time table

2. Course Details

2.1 Course Aim and Summary

The course '**Operating System**' aims at introducing concepts and functions of operating systems. The course emphasises on improving the knowledge on scheduling algorithms and memory management.

2.2 Course Objectives

The objectives of the Course are:

- Study the basic concepts and functions of operating systems.
- Understand the structure and functions of OS.
- Learn about Processes, Threads and Scheduling algorithms.
- Understand the principles of concurrency and Deadlocks.
- Learn various memory management schemes.
- Study I/O management and File systems.
- Learn the basics of Linux system and perform administrative tasks on Linux Servers.

2.3 Course Outcomes

After undergoing this course, students will be able to:

	Outcomes	Bloom's Taxonomy Level
C O 1	Design various Scheduling algorithms	1- Remember
C O 2	Design deadlock, prevention and avoidance algorithms.	2- Understand
C O 3	Compare and contrast various memory management schemes	6- Create
C O 4	Understand the storage management and file structures	3-Apply
C O 5	Understand the functions of operating system.	2- Understand

2.4 Course Content

INTRODUCTION AND SYSTEMS STRUCTURE:

Operating system as an extended machine, resource manager. History of OS – first, second, third, fourth, present. Basic operating system concepts – processes, files, shell. (3)

Operating Systems Structures: operating system services, user operating system interface, system calls and its types, os design and implementation, virtual machines, debugging, system boot.

PROCESS:

Process model – creation, termination, hierarchies, states, implementation. Process synchronization – race conditions, critical sections, mutual exclusion, Peterson's solution, synchronization hardware, semaphores, mutex, monitor, message passing, atomic transactions

THREADS AND SCHEDULING:

Multithreading models, thread libraries, threading issues, threading in java, classic synchronization problems – dining philosophers and readers writers' problems and its programming solutions.

Scheduling: Batch systems, interactive systems, real time systems, threads. Scheduling criteria, scheduling algorithms, thread and multiprocessor scheduling algorithms, examples and algorithms evaluation.

DEADLOCKS MEMORY MANAGEMENT:

Resources, Principles of deadlock, methods for handling deadlock – ostrich and bankers algorithm, detection and recovery, deadlock prevention, deadlock avoidance.

Memory Management: Main memory – swapping, contiguous memory allocation, paging, structure of page table, segmentation, examples. Virtual memory – demand paging, copy on write, page replacement, allocation of frames, thrashing, memory mapped files, allocating kernel memory, memory management utilities

VIRTUALIZATION:

Requirements - Type 1 Hypervisors - Type 2 Hypervisors – Para virtualization - Memory Virtualization - I/O Virtualization - Virtual Appliances - Virtual Machines on Multicore CPUs - Licensing Issues (3)

CASE STUDY: Windows

2.5 Text Book and References Text Book:

1. Silberschatz A, Galvin P and Gagne G “Operating Systems Concepts”, John Wiley & Sons, USA, 2008.
2. Andrew S Tanenbaum, “Operating Systems Design and Implementation”, Prentice Hall of India, New Delhi, 2006.

References:

1. Deitel H M, Deitel J P, David R Choffnes, " Operating Systems", Prentice Hall of India, New Delhi, 2004.
2. Andrew S. Tanenbaum, “Modern Operating Systems”, Prentice Hall of India, New Delhi, 2008
3. Mark Russinovich, David A. Solomon, Alex Ionescu, “Windows Internals: Including Windows Server 2008 and Windows Vista”, Microsoft Press, Cambridge, England, 2009.
4. Gary Nutt, “Operating Systems”, Addison Wesley, USA, 2004.

Course: Object Oriented Analysis and Design using Agile approach

School	School of Engineering
Department	Computer Applications
Programme	B.C.A
Course Title	Object Oriented Analysis and Design using Agile approach
Course Code	20CA2401
Year	2020-2023
Semester	IV
Name of the Faculty Members	
e-mail id	

1. Course Size and Instruction Method

Credits	L	T	P	Hours/Week
4	3	0	2	3

Total Semester hours: 3 x13= 39 **Hours Mentoring Hours:** As per time table

2. Course Details

1.1 Course Aim and Summary

The course '**Object Oriented Analysis and Design Using Agile Approach**' aims at introducing the basics of object oriented analysis and design skills. The course deals with methodologies and modelling of object oriented design. During the course, the students will be taught UML analysis and its design patterns.

1.2 Course Objectives

The objectives of the Course are:

- Learn the basics of OO analysis and design skills
- Learn the UML design diagrams
- Learn to map design to code
- Be exposed to the various testing techniques.

1.3 Course Outcomes

After undergoing this course, students will be able to:

	Outcomes	Bloom's Taxonomy Level
CO1	Design and implement projects using OO concepts	1-Remember

C O 2	Use the UML analysis and design diagrams	2- Understand
C O 3	Apply appropriate design patterns	6- Create
C O 4	To outline and analyze the areas of design pattern correspondence with code.	3-Apply
C O 5	To explain the purpose of applying particular design pattern to a specific module.	2- Understand

1.4 Course Content INTRODUCTION:

Complexity- Structure of complex of systems, Inherent complexity of software, attributes of a complex system, Evolution of object models – Foundations of Object model, Elements of Object model – Major elements: Abstraction, Encapsulation, Modularity and Hierarchy- Minor elements Typing, Concurrency, Persistence.

CLASSES AND OBJECTS:

Nature of an object- Relationships among objects – Nature of class – Relationship among classes – Interplay of Classes and objects- on building quality classes and objects
Classification: Importance of Proper Classification- Identifying classes and objects –Key Abstractions and Mechanisms.

METHODOLOGY AND MODELING:

Object Oriented methodologies - Introduction, Survey of some Object oriented methodologies – Rumbaugh, Booch, Jacobson ,Patterns, Frameworks, Unified approach.

UNIFIED MODELING LANGUAGE:

CLASS DIAGRAM: Notation- Object diagram, Class interface notation.

UML Interaction DIAGRAMS: Sequence Diagrams, Collaboration Diagrams- UML State chart diagram, UML Activity diagram, Implementation diagrams: Component diagram, Deployment diagram.

CASE STUDIES:

Object Oriented Analysis process, Object oriented Design process - Automatic Teller Machine.

2.5 Text Book and References Text Book:

1. Grady Booch, Robert A.Maksimchuk, Michael. W. Engle, Bobbi J. Young, JIM Conallen, Kelli A. Houston “Object Oriented Analysis and Design with Applications”, Pearson Education Inc., USA, 2010

2. Ali Bahrami, "Object Oriented System Development", McGraw Hill International Edition, Singapore, 2008.

References:

1. Rumbaugh J, Blaha M, Premerlani W, Eddy F and Lorenzen W, "Object Oriented Modeling and Design", Prentice Hall of India/ Pearson Education, New Delhi, 2004.
2. Kendall Scott, Martin Fowler, "UML Distilled : A brief guide to the standard Object modeling Language ", Addison Wesley, USA, 2009
3. Atul Kahate, " Object Oriented Analysis and Design ", Tata McGraw-Hill , New Delhi 2007. 4. Sudha Sadasivam G., " Object-Oriented Analysis and Design", Macmillan India, New Delhi, 2009.

Course: Advanced Web Programming

School	School of Engineering
Department	Computer Applications
Programme	B.C.A
Course Title	Advanced Web Programming
Course Code	20CA2402
Year	2020-2023
Semester	IV
Name of the Faculty Members	
e-mail id	

1. Course Size and Instruction Method

Credits	L	T	P	Hours/Week
4	4	0	0	3

Total Semester hours: 3 x13= 39 **Hours Mentoring Hours:** As per time table

2. Course Details

2.5 Course Aim and Summary

The course '**Advanced Web Programming**' aims at introducing computer programming with C language. The course deals with language fundamentals, control flow statements, decision making statements, loops and derived data types like arrays, strings, functions, pointers, structures and unions. The course emphasises on improving problem solving capabilities of students using computers. During the course, the students will be taught to design algorithms, implement and execute problems in the form of C programs.

2.6 Course Objectives

The objectives of the Course are:

- Describe several tools and/or techniques involved in developing professional level Web sites.
- Compare _____ and _____ contrast those tools and/or techniques while _____ analyzing their appropriateness for solving specific problems..

2.7 Course Outcomes

After undergoing this course, students will be able to:

	Outcomes	Bloom's Taxonomy Level
--	----------	------------------------

C O 1	Demonstrate an understanding of the basic principles of the programming process	1- Remember
C O 2	Demonstrate an understanding of the web and related delivery systems.	2- Understand
C O 3	Basics of client side scripting language	6- Create
C O 4	To understand concepts of XML	3-Apply
C O 5	Knowledge on E-Business and its functions	2- Understand

2.8 Course Content Internet Security & HTML:

Overview of internet security, access security, transaction security, security zones, digital IDS, sending / receiving signed & encrypted emails. Introduction to firewalls.

ADVANCE HTML:

Working with images, links and lists, creating tables. working with frames, creating horizontal, vertical frames, named frames, opening new browser window, creating html forms, Adding controls on forms, submitting data from forms, working with multimedia, multimedia sound, video, 3D, Using multimedia files, inline sound and videos. Style sheets: types, creating and, using style sheets.

CSS & Java script & XML:

Introduction to client and server side scripting, introduction to Cascading Style sheet, selectors and there properties, where to include CSS in HTML. Introduction to Java script, data types, operators, conditional statement, loops in Java script, functions, arrays, objects and elements in Java script, form validation using Java script.

XML:

Introduction to XML, Creating XML documents, specifying attributes in DTDs, accessing XML data with XML Data Island, documents. Handling events while loading XML documents.

E-Commerce:

Introduction to E-Business, Electronic Fund Transfer (EFT), Value chain, internet Business strategy, Functional Architecture, implementation Strategies; Building Blocks of E-commerce, System design, creating and managing content etc.; Payment systems; Auxiliary system; transaction Processing; Building e-commerce system, system architecture, secure links etc.; Present and future Trend; Impact of ecommerce; A case Study on development of e-commerce system

2.5 Text Book and References Text Book:

1. Bayross, Web Enable Commercial Application Development Using HTML, DHTML, Javascript, Pen CGI, BPB Publications, 2000
2. J. Jaworski, Mastering Javascript, BPB Publications, 1999

References:

1. T. A. Powell, Complete Reference HTML (Third Edition), TMH, 2002
2. G.Buczek, ASP.NET Developers Guide, TMH, 2002

Course: Database System

School	School of Engineering
Department	Computer Applications
Programme	B.C.A
Course Title	Database System
Course Code	20CA2403
Year	2020-2023
Semester	IV
Name of the Faculty Members	
e-mail id	

1. Course Size and Instruction Method

Credits	L	T	P	Hours/Week
4	3	2	0	3

Total Semester hours: 3 x13= 39 **Hours Mentoring Hours:** As per time table

2. Course Details

2.9 Course Aim and Summary

The course '**Database System**' aims at introducing fundamentals of Database Systems. The course deals with language fundamentals of Transaction Processing and Query Processing. During the course, the students will be taught about relational model , ER diagrams and SQL

2.10

Course Objectives

The objectives of the Course are:

- To expose the students to the fundamentals of Database Management Systems.
- To make the students understand the relational model.
- To familiarize the students with ER diagrams.
- To expose the students to SQL.
- To make the students to understand the fundamentals of Transaction Processing and Query Processing.
- To familiarize the students with the different types of databases.

2.11

Course Outcomes

After undergoing this course, students will be able to:

	Outcomes	Bloom's
--	-----------------	----------------

		Taxonomy Level
C 0 1	Design Databases for applications	1-Remember
C 0 2	Use the Relational model, ER diagrams	2-Understand
C 0 3	Apply concurrency control and recovery mechanisms for practical problems	6- Create
C 0 4	Design the Query Processor and Transaction Processor	3-Apply
C 0 5	Discuss the concepts of index structure and files	2-Understand

2.12

Course Content INTRODUCTION:

Data, Database, Database management system, Characteristics of the database approach, Role of Database administrators, Role of Database Designers, End Users, Advantages of Using a DBMS and When not to use a DBMS.

DBMS Architecture:

Data Models – Categories of data models, Schemas, Instances, and Database state. DBMS Architecture and Data Independence – The Three schema architecture, Data independence. DBMS Languages and Interfaces. Classifications of Database Management Systems

Data Modelling Using Entity:

Relationship Model: Using High Level Conceptual Data Models for Database Design, Example Database applications. Entity types, Entity Sets, Attributes and Keys.

Relationships, Relationship types, Roles and Structural constraints. Weak Entity Types and Drawing E- R Diagrams..

Index Structures for Files:

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

Relational Data Model:

Relation, Integrity constraints - domain, entity and Referential integrity constraints, Basic Relational Algebra operations, select, project and join operations. Functional dependencies and Normalization for Relational Databases, Normalization concepts, first, second, third normal forms, Boyce-Codd normal form. **SQL:** Queries, sub queries, correlated sub query, views, updation of a database through views, Update, Delete.

2.5 Text Book and References Text Book:

1. Abraham Silberschatz, Henry F. Korth, S. Sudharshan, "Database System Concepts", Sixth Edition, Tata McGraw Hill, 2010
2. Ramez Elmasri, Shamkant B. Navathe, "Fundamentals of Database Systems", Sixth Edition, Pearson/ Addison wesley, 2010.

References:

1. O`neil Patric & O`neil Elizabeth, Database Principles, Programming and Performance, 2nd Edition, Margon Kaufmann Publishers Inc.
2. Bipin Desai, "An Introduction to database systems", Galgotia Publications, 1991.

Course: Software Testing

School	School of Engineering
Department	Computer Applications
Programme	B.C.A
Course Title	Software Testing
Course Code	20CA2404
Year	2020-2023
Semester	IV
Name of the Faculty Members	
e-mail id	

1. Course Size and Instruction Method

Credits	L	T	P	Hours/Week
4	3	0	2	3

Total Semester hours: 3 x13= 39 **Hours Mentoring Hours:** As per time table

2. Course Details

2.1 Course Aim and Summary

The course '**Software Testing**' aims at introducing different test cases and design of it. The course deals with language test management and its techniques.

2.2 Course Objectives

The objectives of the Course are:

- Expose the criteria for test cases.
- Learn the design of test cases.
- Be familiar with test management and test automation techniques.
- Be exposed to test metrics and measurements.

2.3 Course Outcomes

After undergoing this course, students will be able to:

	Outcomes	Bloom's Taxonomy Level
CO1	Design test cases suitable for a software development for different domains	1- Remember
CO2	Identify suitable tests to be carried out	2- Understand
CO3	Prepare test planning based on the document	6- Create
CO4	Document test plans and test cases designed	3-Apply
CO5	Develop and validate a test plan	2- Understand

2.4 Course Content

Introduction to Software Quality Engineering:

Software Quality -Software Quality Assurance -Reviews, Inspections and Walkthroughs.

Introduction to Testing:

Guiding Principles of testing – Composition of a testing team – Essential Skills of a tester – Types of Testing – Evaluating the quality of test cases – Techniques for reducing number of test cases-

Requirements for effective testing – Test Oracle – Economics of testing – Handling Defects

White Box (Structural) Testing:

Introduction to control paths – Control flow Testing – Basis Path testing – Linear Code Sequence and Jump Coverage – Looping testing – Data Flow Testing – Slice-based Testing – Pitfalls of White box testing – Tools for White box testing.

Integration Testing:

Types of integration testing – Functional Testing – Non- Functional Testing – Acceptance Testing- Regression Testing.

TEST Management:

Activities in test management – Evaluation of test Effectiveness – Release management – Tools in test management. Cloud Testing – Test Automation.

2.5 Text Book and References Text Book:

1. Anirban Basu , “Software Quality Assurance, Testing and Metrics “, PHI.

Course: Internet and Web Architecture

School	School of Engineering
Department	Computer Applications
Programme	B.C.A
Course Title	Internet and Web Architecture
Course Code	20CA2408
Year	2020-2023
Semester	IV
Name of the Faculty Members	
e-mail id	

1. Course Size and Instruction Method

Credits	L	T	P	Hours/Week
4	2	2	2	3

Total Semester hours: 3 x13= 39 Hours **Mentoring Hours:** As per time table

2. Course Details

2.1 Course Aim and Summary

The course '**Internet and Web Architecture**' aims at introducing client side and server side programming. The course deals with basic knowledge on JSP programming with its classes and library files.

2.2 Course Objectives

The objectives of the Course are:

- To understand the client / server programming
- To learn CSS to implement a variety of presentation effects using HTML and XML documents.
- To know the unique features of scripting languages

2.3 Course Outcomes

After undergoing this course, students will be able to:

	Outcomes	Bloom's Taxonomy Level
C O 1	Upon Completion of the course, the students should be able to	1- Remember
C O 2	Create web pages using HTML, JavaScript and CSS	2- Understand
C O 3	Understand the key technologies of Internet	6- Create

C O 4	Distinguish the concepts of Java script and Java servlet	3-Apply
C O 5	Understand the web service technologies	2- Understand

2.4 Course Content Web Essentials:

Internet – Web clients – Web servers – Markup languages – XHTML 1.0 – Cascading Style Sheets (CSS): Features- Style rule cascading and inheritance - Text properties – CSS box model.

Client Side Programming:

Java script operators - Java script objects – Arrays – Build-in objects - DOM: History and levels- Document tree- DOM event handling- Non compliant browsers.

Server Side Programming:

Java servlet: Architecture – Servlet life cycle – Parameter data – Sessions – Cookies – Other servlet capabilities – Data storage –Servlet and concurrency

JSP Programming:

Introduction to Java Server Pages – JSP and servlets – Running JSP applications - Basic JSP – Java beans classes and JSP - Tag libraries and files.

Web Services:

Web services concepts - Writing java web services – Web services for clients – WSDL – Representing data types:XML schema – Communicating object data: SOAP – SOAP encoding of struct data.

2.5 Text Book and References Text Book:

1. Jeffrey C Jackson, Web Technology – A computer Science perspective, Person Education, 2009.
2. Chris Bates, Web Programming – Building Internet Applications, Wiley India, 2006.
3. Gopalan. N.P, Web Technology A Developer Perspectives, PHI, 2010.

References

1. Deitel, Deitel and Neito, INTERNET and WORLD WIDE WEB – How to program, Pearson education, 2011.

SEMESTER/YEAR : 5TH SEM / 3RD YEAR
COURSE CODE : 19CA3501
TITLE OF THE COURSE: INFORMATION AND NETWORK SECURITY L: T/A: P: C: 2: 1: 0: 3

COURSE OBJECTIVES:

1. To understand the fundamentals of Cryptography
2. To acquire knowledge on standard algorithms used to provide confidentiality, integrity and authenticity.
3. To understand the various key distribution and management schemes.
4. To understand how to deploy encryption techniques to secure data in transit across data networks
5. To design security applications in the field of Information technology

COURSE OUTCOMES: At the end of the course students will be able to:

1. Implement basic security algorithms required by any computing system.
2. Analyze the vulnerabilities in any computing system and hence be able to design a security solution.
3. Analyze the possible security attacks in complex real time systems and its effective countermeasures.
4. Identify the security issues in the network and resolve it.
5. Evaluate security mechanisms using rigorous approaches, including theoretical derivation, modeling, and simulations
6. Formulate research problems in the computer security field

MODULE 1 - INTRODUCTION 9hrs

An Overview of Computer Security-Security Services-Security Mechanisms-Security Attacks-Access Control Matrix, Policy-Security policies, Confidentiality policies, Integrity policies and Hybrid policies.

MODULE 2 - CRYPTOSYSTEMS & AUTHENTICATION 9hrs

Classical Cryptography-Substitution Ciphers-permutation Ciphers-Block Ciphers - DES Modes of Operation- AES-Linear Cryptanalysis, Differential Cryptanalysis- Hash Function - SHA 512-Message Authentication Codes-HMAC - Authentication Protocols

MODULE 3 - PUBLIC KEY CRYPTOSYSTEMS 9 hrs

Introduction to Public key Cryptography- Number theory- The RSA Cryptosystem and Factoring Integer- Attacks on RSA-- Digital Signature Algorithm-Finite Fields-Elliptic Curves Cryptography- Key management – Session and Interchange keys, Key exchange and generation-PKI

MODULE 4 - SYSTEM IMPLEMENTATION

9 hrs

Design Principles, Representing Identity, Access Control Mechanisms, Information Flow and Confinement Problem Secure Software Development: Secured Coding - OWASP/SANS Top Vulnerabilities - Buffer Overflows - Incomplete mediation - XSS - Anti Cross Site Scripting Libraries - Canonical Data Format - Command Injection - Redirection - Inference - Application Controls.

MODULE 5 - NETWORK SECURITY 9 hrs

Secret Sharing Schemes-Kerberos- Pretty Good Privacy (PGP)-Secure Socket Layer (SSL)- Intruders - HIDS- NIDS - Firewalls - Viruses.

Text Books

1. William Stallings, "Cryptography and Network Security: Principles and Practices", Third Edition, Pearson Education, 2006.
2. Matt Bishop, "Computer Security art and science", Second Edition, Pearson Education, 2002

Reference Books

1. Wade Trappe and Lawrence C. Washington, "Introduction to Cryptography with Coding Theory" Second Edition, Pearson Education, 2007
2. Jonathan Katz, and Yehuda Lindell, Introduction to Modern Cryptography, CRC Press, 2007
3. Douglas R. Stinson, "Cryptography Theory and Practice", Third Edition, Chapman & Hall/CRC, 2006
4. Wenbo Mao, "Modern Cryptography - Theory and Practice", Pearson Education, First Edition, 2006.
5. Network Security and Cryptography, Menezes Bernard, Cengage Learning, New Delhi, 2011
6. Man, Young Rhee, Internet Security, Wiley, 2003

SEMESTER/YEAR : 5TH SEM / 3RD YEAR
COURSE CODE : 19CA3502
TITLE OF THE COURSE: BIG DATA ANALYTICS USING HADOOP
L: T/A: P: C: 2: 1: 2: 4

COURSE OBJECTIVES:

1. To Learn business case studies for big data analytics
2. To Understand NoSQL big data management
3. To manage Big data without SQL
4. To understanding map-reduce analytics using Hadoop and related tools

COURSE OUTCOMES: At the end of the course students will be able to:

1. Describe big data and use cases from selected business domains
2. Explain NoSQL big data management
3. Install, configure, and run Hadoop and HDFS
4. Perform map-reduce analytics using Hadoop
5. Use Hadoop related tools such as HBase, Cassandra, Pig, and Hive for big data Analytics

MODULE 1 - UNDERSTANDING BIG DATA

9hrs

What is big data – why big data – Data Storage and Analysis, Comparison with Other Systems, Rational Database Management System, Grid Computing, Volunteer Computing, convergence of key trends – unstructured data – industry examples of big data –big data and healthcare – big data in medicine – advertising and big data.

MODULE 2 - NOSQL DATA MANAGEMENT

9hrs

Introduction to NoSQL – aggregate data models – aggregates – key-value and document data models – relationships –graph databases – schema less databases – materialized views – distribution models – sharding – version – Map reduce –partitioning and combining – composing map-reduce calculations

MODULE 3 - BASICS OF HADOOP 9 hrs

Data format – analyzing data with Hadoop – scaling out – Hadoop streaming – Hadoop pipes – design of Hadoop distributed file system (HDFS) – HDFS concepts – Java interface – data flow – Hadoop I/O – data integrity – compression– serialization – Avro – file-based data structures.

MODULE 4 - MAPREDUCE APPLICATIONS

9 hrs

MapReduce workflows – unit tests with MR Unit – test data and local tests – anatomy of MapReduce job run – classic Map-reduce – YARN – failures in classic Map-reduce and YARN – job scheduling – shuffle and sort – task execution –MapReduce types – input formats – output formats

MODULE 5 - HADOOP RELATED TOOLS

9 hrs

HBase – data model and implementations – HBase clients – Hbase examples –praxis. Cassandra – Cassandra data model –Cassandra examples – Cassandra clients –Hadoop integration. Pig – Grunt – pig data model – Pig Latin – developing and testing Pig Latin scripts. Hive – data types and file formats – HiveQL data definition – HiveQL data manipulation –HiveQL queries.

Text Books

1. Tom White, "Hadoop: The Definitive Guide", Third Edition, O'Reilley, 2012.
2. Eric Sammer, "Hadoop Operations", O'Reilley, 2012.

Reference Books

1. Vignesh Prajapati, Big data analytics with R and Hadoop, SPD 2013.
2. E. Capriolo, D. Wampler, and J. Rutherglen, "Programming Hive", O'Reilley, 2012.
3. Lars George, "HBase: The Definitive Guide", O'Reilley, 2011.
4. Alan Gates, "Programming Pig", O'Reilley, 2011.

SEMESTER/YEAR : 5TH SEM / 3RD YEAR
COURSE CODE : 19CA3505
TITLE OF THE COURSE: MOBILE PROGRAMMING
L: T/A: P: C

: 2: 1: 2: 4

COURSE OBJECTIVES:

1. Understand the basic concepts of mobile computing
2. Be familiar with the network protocol stack
3. Learn the basics of mobile telecommunication system
4. Be exposed to Ad-Hoc networks
5. Gain knowledge about different mobile platforms and application Development.

COURSE OUTCOMES: At the end of the course students will be able to:

1. Acquire technical competency and skills in developing applications using Android.

MODULE 1 - INTRODUCTION 9hrs

Characteristics and advantages of mobile communication, types of mobile applications – development approaches, overview of mobile strategy and designing mobile solutions. Mobile computing project structure, building and testing. Evolution of Modern Mobile Wireless communication system. Mobile computing vs. wireless - MAC Protocols –Wireless MAC Issues – Fixed Assignment Schemes – Random Assignment Schemes – Reservation Based Schemes.

MODULE 2 - MOBILE PLATFORMS AND APPLICATIONS 9hrs

Mobile Device Operating Systems – Special Constrains & Requirements – Commercial Mobile Operating Systems – Software Development Kit: iOS, Android, BlackBerry, Windows Phone –M-Commerce – Structure – Pros & Cons – Mobile Payment System.

MODULE 3 - INTRODUCTION TO ANDROID 9 hrs

Android: Introduction, trends, platforms, Android Development Setup like, Android Studio, Eclipse, Android SDK, tools. Emulator setup. App Behaviour on the Android Runtime (ART). Platform Architecture. Application framework and basic App Components resources.

MODULE 4 - MOBILE APP DEVELOPMENT USING ANDROID 9 hrs

Android App Use Case (Use Case Diagram (UML))
Implement user Interface - layout, values, asset XML representation, generated R.java file, Android manifest file. Activities, Intent and UI Design - activities life-cycle. Android Components – layouts, fragments, basic views, list views, picker views, adapter views, Menu, Action Bar etc. - Managing data using SQLite database.

MODULE 5 - DATA PERSISTENCE AND GOOGLE APIs FOR ANDROID 9 hrs

Managing data using SQLite, user content provider and persisting data into database Google APIs for Android - Maps, Cloud Messaging, Authentication, Storage, Hosting and Google Play services.

Text Books

1. Learning Android by Marko Gargenta O'reilly Publications
2. Professional Android™ Application Development by Reto Meier, Wrox Publication
3. Head First Android Development by Jonathan Simon O'reilly Publications

Reference Books

1. Jochen H. Schller, "Mobile Communications", Second Edition, Pearson Education, New Delhi, 2007.
2. Dharma Prakash Agarval, Qing and An Zeng, "Introduction to Wireless and Mobile systems", Thomson Asia Pvt Ltd, 2005.
3. Uwe Hansmann, Lothar Merk, Martin S. Nicklons and Thomas Stober, "Principles of Mobile Computing", Springer, 2003

SEMESTER/YEAR : 6TH SEM / 3RD YEAR

COURSE CODE : 19CA3608

TITLE OF THE COURSE: CLOUD COMPUTING

L: T/A: P: C

: 2: 2: 2: 4

COURSE OBJECTIVES:

1. To understand the various issues in cloud computing
2. To familiarize themselves with the lead players in cloud
3. To appreciate the emergence of cloud as the next generation computing paradigm
4. To be able to set up a private cloud

COURSE OUTCOMES: At the end of the course students will be able to:

1. Articulate the main concepts, key technologies, strengths and limitations of cloud computing
2. Identify the architecture, infrastructure and delivery models of cloud computing
3. Explain the core issues of cloud computing such as security, privacy and interoperability.
4. Choose the appropriate technologies, algorithms and approaches for the related issues

MODULE 1 - INTRODUCTION 9hrs

Evolution of Cloud Computing -System Models for Distributed and Cloud Computing - NIST Cloud Computing Reference Architecture -IaaS - On-demand Provisioning - Elasticity in Cloud - Examples of IaaS Providers - PaaS - Examples of PaaS Providers - SaaS - Examples of SaaS Providers - Public, Private and Hybrid Clouds – Google App Engine, Amazon AWS - Cloud Software Environments -Eucalyptus, Open Nebula, Open Stack, Nimbus.

MODULE 2 - VIRTUALIZATION 9hrs

Basics of Virtual Machines - Process Virtual Machines – System Virtual Machines – Emulation – Interpretation – Binary Translation - Taxonomy of Virtual Machines. Virtualization – Management Virtualization – Hardware Maximization – Architectures – Virtualization Management – Storage Virtualization – Network Virtualization.

MODULE 3 - VIRTUALIZATION INFRASTRUCTURE 9 hrs

Comprehensive Analysis – Resource Pool – Testing Environment –Server Virtualization – Virtual Workloads – Provision Virtual Machines –Desktop Virtualization – Application Virtualization – Work with AppV – Mobile OS for smart phones – Mobile Platform Virtualization – Collaborative Applications for Mobile platforms

MODULE 4 - PROGRAMMING MODEL 9 hrs

Map Reduce Hadoop Distributed File Systems – Hadoop I/O – Developing Map Reduce Applications – Working of Map Reduce – Types and Formats – Setting up Hadoop Cluster

MODULE 5 - CLOUD INFRASTRUCTURE AND SECURITY

9 hrs

Architectural Design of Compute and Storage Clouds - Inter Cloud Resource Management - Resource Provisioning and Platform Deployment - Global Exchange of Cloud Resources - 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.

Text Books

1. Kai Hwang, Geoffrey C Fox, Jack G Dongarra, "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann Publishers, 2012.
2. John W.Rittinghouse and James F.Ransome, "Cloud Computing: Implementation, Management, and Security", CRC Press, 2010.
3. Jim Smith, Ravi Nair, "Virtual Machines: Versatile Platforms for Systems and Processes", Elsevier/Morgan Kaufmann, 2005.
4. Danielle Ruest, Nelson Ruest, "Virtualization: A Beginner's Guide", McGraw-Hill Osborne Media, 2009.
5. Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi, "Mastering Cloud Computing", Tata McGraw Hill, 2013

SEMESTER/YEAR : **II SEM / I YEAR**
COURSE CODE : **19CA1204**
TITLE OF THE COURSE : **Computer Organization II**
L: T: P: S/P: C : **03-00-00-00-03**

Course Objectives : This course aims to equip the student with assembly level programming using 8086 microprocessor.

Course outcomes : At the end of the course student will be able
 4. to understand the design the data part and control part of a processor
 5. to understand memory technology, I/O systems and I/O operation and use them in the design of a computing system
 6. To understand and appreciate the 8086 processor and program the same to solve problems.

Module 1	Introduction to the microprocessor and computer: Internal processor architecture, Functional block diagram, Bus, Clock signals, addressing modes.	10 hrs
Module 2	Programming 8086: Data movement instructions, Arithmetic and logic instructions, Program control instructions, string instructions, programming techniques, examples. Modular programming: stacks, subroutines, Macros.	10 hrs
Module 3	Assembler Directives: Data Definition and Storage Allocation, Program Organization, Alignment, Program End, Value Returning Attribute, Procedure Definition, Macro Definition, Data Control, Branch Displacement, Header File, Inclusion, Target Machine Code, Generation Control Directives.	10 hrs
Module 4	Interrupts and interrupt routines, I\O interface and programming: Fundamental I\O considerations, Data transfer schemes9Programmed I\O, Interrupt I\O, DMA, System bus structure, Min.Max Modes, Application of 8259, 8255,8251,8257,8253	10 hrs
Module 5	Introduction to other 16 bit 32 bit processors: 80286, 386, 486, Pentium and Pentium Pro-processor.	10 hrs
Text Books	1. YuChang Liu & Glenn A Gibson, "Microcomputer systems: the 8086\8088 Family: Architecture, Programming and design", PHI 2. Microprocessor 8086 : Architecture, Programming and Interfacing : Mathur Sunil, PHI Learning	

SEMESTER/YEAR : **I SEM / I YEAR COURSE**
CODE : **20CA1101 TITLE OF**
THE COURSE : **MATHEMATICS-I**
L: T: P: S/P: C : **03-01-00-00-04**

Course Objectives The Curriculum supports the prerequisites to enhance their Mathematical knowledge towards their understanding mathematical Concepts and help them to persuade research work in concerned fields with the help of Mathematical approach.

Course outcomes At the end of the course student will be able to
 1. understand and use counting principle, use Propositional calculus in Theorem proving
 2. use suitable algebraic structures to model the given scenario/system
 3. use constructions used in proofs as algorithms

Module 1 **MATRICES** 09 hrs
 Definition, Types of Matrices, Addition, Subtraction, Scalar Multiplication and Multiplication

of Matrices, Adjoint, Inverse, Eigen values and Eigen Vectors of a Matrix, Caley-Hamilton

Theorem (Statement only) Rank of a matrix, Row reduced echelon form and normal form

Solution of homogeneous and nonhomogeneous system of equations.

Practical's: Solving problems under matrices using FOSS TOOLS SCILAB/Maxima

DETERMINANTS:

Definition, Minors, Cofactors, Properties of Determinants Cramer's rule

Module 2 **SETS AND RELATIONS** 09 hrs

SETS:

Sets, Subsets, Types of Sets, Operation on Sets, Applications.

RELATIONS AND FUNCTIONS:

Definition, Types of functions, Types of relations with illustrations and graphs

Module 3 **DIFFERENTIAL CALCULUS** 09 hrs

Limit, Continuity, Differentiation, Product rule Quotient rule Successive differentiation,

Leibnitz theorem for finding nth derivative of product of functions(only statement) Partial

derivatives, homogeneous functions Euler's theorem (only statement) maxima and minima,

Taylor's series and Maclaurin's Series (without proof).

Practical's: Solutions Using FOSS TOOLS SCILAB/MAXIMA

Module 4 **DIFFERENTIAL EQUATIONS** 08 hrs

Different types of differential equations, solving different equations

ALGEBRAIC STRUCTURES 10 hrs

Module 5 Introduction, operations, semi-groups, groups, subgroups, normal subgroups, isomorphism

and homomorphism, rings, integral domains and fields, Polynomials over a field.

SEMESTER/YEAR : **I SEM / I YEAR**
COURSE CODE : **20CA1103**
TITLE OF THE COURSE : **COMPUTER ORGANIZATION - I**
L: T: P: S/P: C : **03-00-00-00-03**

Course Objectives : This course aims at providing an overview of basic digital design techniques, basic structure of a digital computer and its organization.

Course outcomes : At the end of the course student will be able

6. to demonstrate knowledge of binary number theory, Boolean algebra and binary codes.
7. to analyse and design combinational systems using
8. standard gates and minimization methods (such as Karnaugh maps,
9. Quine-McCluskey Algorithm).

To analyse and design combinational systems composed of Standard combinational modules, such as multiplexers and decoders.

10. to analyse the given problems and design sequential Circuits

Module 1	INTRODUCTION TO DIGITAL COMPUTER Concept of Digital Computer, Types of Software – System software / Application software / Utility Software. Compilers, Interpreters, Assemblers, Linker, Loader. System Bus Structure	06 hrs
Module 2	NUMBER SYSTEM AND LOGIC DESIGN MINIMIZATION TECHNIQUES Introduction to number system, Signed Binary Representation, Logic gates and Boolean Algebra.	06 hrs
Module 3	COMBINATIONAL AND SEQUENTIAL LOGIC CIRCUITS Minimization Techniques and circuit design, Codes and Arithmetic operation, Combinational Circuits Design	06 hrs
Module 4	COMBINATIONAL AND SEQUENTIAL LOGIC CIRCUITS(CONTD) Multiplexers, De-multiplexers, decoders, Sequential circuit Design Counter and registers	06 hrs
Module 5	INPUT/ OUTPUT AND MEMORY ORGANIZATION OF COMPUTER I/O Devices, Memory Systems, Interrupts and instruction cycle, Addressing modes	06 hrs

SEMESTER/YEAR : **I SEM / I YEAR**
COURSE CODE : **20CA1107**
TITLE OF THE COURSE : **PROGRAMMING PRACTICALS**

Course Objectives The objective of the subject is to teach the student the basics of Programming starting with simple problem solving techniques implemented using Python language

Course outcomes At the end the course the students will

3. understand Programming concepts, building blocks of a program, different constructs file i/o operations
4. Be able to design using advanced concepts such as functions, modules and regular expressions in designing programs.

BUILDING BLOCKS

Branching: Decision making with IF statement, IF-ELSE statement, Nested IF statement, ELSE-IF ladder, switch statement, goto statement. break, continue statement

Iterative statements: For, while, and do-while loop, jumps in loops. Arrays, pointers and structures.

10 hrs

SEQUENCES , FILES AND DICTIONARIES

Sequences, FILE I/O Operations, dictionaries and sets

11 hrs

PROGRAMMING CONCEPTS

10 hrs

Functions , modules and regular expressions

- Text Books**
1. Programming in Python 3 (Second Edition) A Complete Introduction to the Python Language by Mark Summerfield
 4. Head First Python – Paul Barry
 5. Gottfried, Byron S., Programming with C, Tata McGraw Hill

ReferenceBooks

3. Problem Solving With C: Jones and Harrow, WILEY INDIA PVT LTD.
4. An Introduction to Programming and Problem Solving With Pascal, G. Michael Schneider Steven W. Weingart David M. Perlman, WILEY INDIA PVT LTD.

SOFTWARES C/PYTHON

SEM/YEAR : II SEM

COURSE CODE : 16CA107

TITLE OF THE COURSE : PRINCIPLES OF PROGRAMMING

L: T/A:P: C : 4 : 0 : 0 : 4

Course Objectives

This subject covers the basics of OOPS concepts such as Inheritance, Polymorphism, Event driven programming, Exceptions and Concurrent programming.

Course outcomes

The students will

- i) Understand OOP concepts, programming blocks. Packages and interfaces.
- ii) Understand and use the exception handling and design programs with multithreading.

Module 1

Basics of Object Oriented Programming (OOP) :Need for OO paradigm, Classes and objects, Constructors and Destructors, Access control, Inheritance, Abstraction, Encapsulation, Polymorphism, Overriding, String handling. 12 Hours

Module 2

Programming blocks :Data types , variables, Identifiers, Key words, scope and life time of variables, operators, expressions, control statements, type conversion and casting.

12Hours

Module 3

Packages and Interfaces: Defining, Creating and Accessing a Package, importing packages, Interfaces, abstract Classes, implementing interface, variables in interface, extending interfaces. 12 Hours

Module 4

Exception handling and Multithreading : Concepts of exception handling, benefits of exception handling, Termination or presumptive models, exception hierarchy, built in exceptions, creating own exception sub classes. Thread life cycle, creating threads, synchronizing threads, daemon threads, and thread groups. 12 Hours

Text Books

1. Java: How to Program, 8/e, Dietal, Dietal, PHI
2. The C++ Programming Language , Bjarne Stroustrup
3. Java: The complete reference, 7/e, Herbert Schildt, TMH.

SEM/YEAR : IV SEM

COURSE CODE : 16CA207

TITLE OF THE COURSE : OBJECT ORIENTED ANALYSIS AND DESIGN USING UML

L: T/A:P: C : 3 : 2 : 0 : 4

Course Objectives

- ☒ To understand the fundamentals of objects and their modeling
- ☒ To differentiate unified process from other approaches
- ☒ To emphasize on modeling based software design
- ☒ To familiarize with the modeling languages
- ☒ To reinforce software design with design patterns

Course outcomes

At the end of the course student will be able

- ☒ To express software design with UML diagrams
- ☒ To analyze the communication of software modules using interaction diagrams
- ☒ To identify and map basic software requirements in UML modeling
- ☒ To be capable of transforming UML based software design into pattern based design framework using design patterns
- ☒ To explain the purpose of applying particular design pattern to a specific module
- ☒ To outline and analyze the areas of design pattern correspondence with code

Module 1

INTRODUCTION: **Complexity- Structure of complex of systems**, Inherent complexity of software, attributes of a complex system, Evolution of object models – Foundations of Object model, Elements of Object model –

Major elements: Abstraction, Encapsulation, Modularity and Hierarchy- Minor elements

Typing, Concurrency, Persistence. 09 hrs

Module 2

CLASSES AND OBJECTS – **Nature of an object- Relationships among objects** – Nature of class – Relationship among classes – Interplay of Classes and objects- on building quality classes and objects Classification: Importance of Proper Classification- Identifying classes and objects –Key Abstractions and Mechanisms. 09 hrs

Module 3

METHODOLOGY AND MODELING: **Object Oriented methodologies** - Introduction, **Survey of some Object oriented methodologies** – Rumbaugh, Booch, Jacobson ,Patterns, Frameworks, Unified approach. 09 hrs

Module 4

UNIFIED MODELING LANGUAGE:

CLASS DIAGRAM: Notation- Object diagram, Class interface notation. UML Interaction DIAGRAMS: Sequence Diagrams, Collaboration Diagrams- UML State chart diagram, UML Activity diagram, Implementation diagrams: Component diagram, Deployment diagram
09 hrs

Module 5

CASE STUDIES: Object Oriented Analysis process, Object oriented Design process - Automatic Teller Machine 09 hrs

Text Books

1. Grady Booch, Robert A.Maksimchuk, Michael. W. Engle, Bobbi J. Young, JIM Conallen, Kelli A. Houston “Object Oriented Analysis and Design with Applications”, Pearson Education Inc., USA, 2010
2. Ali Bahrami, “Object Oriented System Development”, McGraw Hill International Edition, Singapore, 2008.

Reference Books

1. Rumbaugh J, Blaha M, Premerlani W, Eddy F and Lorenzen W, "Object Oriented Modeling and Design", Prentice Hall of India/ Pearson Education, New Delhi, 2004.
2. Kendall Scott, Martin Fowler, "UML Distilled : A brief guide to the standard Object modeling Language ", Addison Wesley, USA, 2009
3. AtulKahate, " Object Oriented Analysis and Design ", Tata McGraw-Hill , New Delhi 2007.
4. SudhaSadasivam G., " Object-Oriented Analysis and Design", Macmillan India, New Delhi, 2009.

SEM/YEAR : IV SEM

COURSE CODE : 16CA225

TITLE OF THE COURSE : COMPUTER GRAPHICS AND MULTIMEDIA

L: T/A:P: C : 2 : 2 : 2 : 4

Course Objectives

Gain knowledge about graphics hardware devices and software used.

- ☑ Understand the two dimensional graphics and their transformations.
- ☑ Understand the three dimensional graphics and their transformations.
- ☑ Appreciate illumination and color models.
- ☑ Be familiar with understand clipping techniques.

Course outcomes

At the end of the course, the student should be able to:

- ☑ Design and Apply of two dimensional transformations.
- ☑ Design and Apply of three dimensional transformations.
- ☑ Apply clipping techniques to graphics.
- ☑ Design animation sequences.

Module 1

OUTPUT PRIMITIVES: Introduction – Points and Lines – DDA Algorithm – Circle-Generating Algorithms – Ellipse-Generating Algorithms – Parallel Curve Algorithms – Curve Functions – Attributes. 09 hrs

Module 2

TWO-DIMENSIONAL GEOMETRIC TRANSFORMATIONS & VIEWING: Basic Transformation – Matrix Representations – Composite Transformation – Other Transformations – Affine Transformation – The viewing Pipeline – Clipping Operation – Point Clipping – Line Clipping – Polygon Clipping – Curve Clipping – Text Clipping – Exterior Clipping 09 hrs

Module 3

INTRODUCTION: Introduction – Multimedia applications – Multimedia System Architecture – Evolving technologies for Multimedia – Defining objects for Multimedia systems – Multimedia Data interface standards – Multimedia Databases.

MULTIMEDIA COMPRESSION :Types of Compression – Binary Image Compression Schemes – Color, Gray Scale, And Still-video Image compression – Video Image Compression – Audio Compression . 09 hrs

Module 4

MULTIMEDIA DATA & FILE FORMAT STANDARDS: Rich-Text Format – TIFF - RIFF – MIDI – JPEG – AVI – MPEG- TWAIN- Multimedia I/O technologies - Digital voice and audio – Video image and animation – Full motion video – Storage and retrieval Technologies.

MULTIMEDIA AUTHORING, USER INTERFACE & HYPERMEDIA: Multimedia Authoring Systems – Hypermedia Application Design Consideration – User Interface Design - Hypermedia messaging - Mobile Messaging – Hypermedia message component – Creating Hypermedia message – Integrated multimedia message standards – Integrated Document management 09 hrs

Module 5

COLOR MODELS & COMPUTER ANIMATION: Properties of Light – Standard Primaries – Intuitive Color Concepts – Color models – Color Selection and Application – Computer Animation: Design of Animation Sequences – Raster Animation – Key-Frame Systems Morphing – Motion Specifications – Virtual Reality case study. 09 hrs

Text Books

1. Donald Hearn and Pauline Baker M, "Computer Graphics C Version", Pearson Education, New Delhi, 2003.
2. Foley, Vandam, Feiner and Huges, "Computer Graphics: Principles & Practice", Pearson Education, Asia, 2003.

Reference Books

1. Prabat K Andleigh and Kiran Thakrar, "Multimedia Systems Design", Prentice Hall of India, New Delhi 2003.
2. Judith Jeffcoate, "Multimedia in practice technology and Applications", Prentice Hall of India, New Delhi, 2009.

SEMESTER/YEAR : **I SEM / I YEAR**
COURSE CODE : **19CA1172**
TITLE OF THE COURSE : **Practicals - COMPUTER ORGANIZATION - I**
L: T: P: S/P: C : **03-00-00-00-03**

Course Objectives This course aims at providing an overview of basic digital design techniques, basic structure of a digital computer and its organization.

Course outcomes At the end of the course student will be able

1. to demonstrate knowledge of binary number theory, Boolean algebra and binary codes.
2. to analyse and design combinational systems using
3. standard gates and minimization methods (such as Karnaugh maps,
4. Quine-McCluskey Algorithm).

To analyse and design combinational systems composed of Standard combinational modules, such as multiplexers and decoders.

5. to analyse the given problems and design sequential Circuits

COMBINATIONAL AND SEQUENTIAL LOGIC CIRCUITS 06 hrs
 Minimization Techniques and circuit design, Codes and Arithmetic operation, Combinational Circuits Design

COMBINATIONAL AND SEQUENTIAL LOGIC CIRCUITS(CONTD) 06 hrs
 Multiplexers, De-multiplexers, decoders, Sequential circuit Design Counter and registers

INPUT/ OUTPUT AND MEMORY ORGANIZATION OF COMPUTER 06 hrs
 I/O Devices, Memory Systems, Interrupts and instruction cycle, Addressing modes

SEMESTER/YEAR : **II SEM / I YEAR**
COURSE CODE : **19CA1273**
TITLE OF THE COURSE: Practicals - Computer Organization II
L: T: P: S/P: C : **03-00-00-00-03**

Assembler Directives:

Data Definition and Storage Allocation, Program Organization, Alignment, Program End, Value Returning Attribute, Procedure Definition, Macro Definition, Data Control, Branch Displacement, Header File, Inclusion, Target Machine Code, Generation Control Directives.

Interrupts and interrupt routines, I/O interface and programming: Fundamental I/O considerations, Data transfer schemes9Programmed I/O, Interrupt I/O, DMA, System bus structure, Min.Max Modes, Application of 8259, 8255,8251,8257,8253

Introduction to other 16 bit 32 bit processors: 80286, 386, 486, Pentium and Pentium Pro-processor.

SEMESTER/YEAR : **II SEM / I YEAR**
COURSE CODE : **16CA108**
TITLE OF THE COURSE : **DATA STRUCTURES**
L: T: P: S/P: C : **03-00-00-00-03**

Course objectives The objective of the subject is to teach the student the usage of data structures using any Programming language.

Course outcomes At the end of the course student will be able
4. To understand and explain linear and dynamic allocation of memory using array and linked list
5. To design programs for queue, stacks, sorting, Searching using linear data structures.
6. To use Nonlinear data structures to create Tree, Graphs.

Module 1	Searching Techniques Preliminaries of algorithm, Algorithm analysis and complexity. Recursion: Definition, Design Methodology and Implementation of recursive algorithms, Linear and binary recursion, recursive algorithms for factorial function, GCD computation, Fibonacci sequence, Towers of Hanoi, Tail recursion List Searches using Linear Search, Binary Search, Fibonacci Search.	10 Hour
Module 2	Sorting Techniques Basic concepts, Sorting by : insertion (Insertion sort), selection (heap sort), exchange (bubble sort, quick sort), distribution (radix sort) and merging (merge sort) Algorithms.	10 Hour
Module 3	Stacks and Queues Basic Stack Operations, Representation of a Stack using Arrays, Stack Applications: Reversing list, Factorial Calculation, In-fix- to postfix Transformation, Evaluating Arithmetic Expressions. Queues: Basic Queues Operations, Representation of a Queue using array, Implementation of Queue Operations using Stack, Applications of Queues-Round robin Algorithm, Enqueue, Dequeue, Circular Queues, Priority Queues.	10 Hour
Module 4	Linked Lists Introduction, single linked list, representation of a linked list in memory, Operations on a single linked list, merging two single linked lists into one list, Reversing a single linked list, applications of single linked list to represent polynomial expressions and sparse matrix manipulation, Advantages and disadvantages of single linked list, Circular linked list, Double linked list	11 Hour
Module 5	Trees and advance Data Structures Basic tree concepts, Binary Trees: Properties, Representation of Binary Trees using arrays and linked lists, operations on a Binary tree , Binary Tree Traversals (recursive), Creation of binary tree from in-order and pre (post) order traversals. Adv Data Structures	10 Hour
Text Books	1. Data Structures, 2/e, Richard F, Gilberg , Forouzan, Cengage 2. Data Structures and Algorithms, 2008,G.A.V.Pai, TMH	
Reference Books	1. Data Structure with C, Seymour Lipschutz, TMH	

SEMESTER/YEAR : I SEM / I YEAR
COURSE CODE : 16CA171
TITLE OF THE COURSE : FUNDAMENTALS OF
PROGRAMMING PRACTICALS

Course Objectives The objective of the subject is to teach the student the basics of Programming starting with simple problem solving techniques implemented using Python language

Course outcomes At the end the course the students will

5. understand Programming concepts, building blocks of a program, different constructs file i/o operations
6. Be able to design using advanced concepts such as functions, modules and regular expressions in designing programs.

BUILDING BLOCKS

Branching: Decision making with IF statement, IF-ELSE statement, Nested IF statement, ELSE-IF ladder, switch statement, goto statement. break, continue statement

Iterative statements: For, while, and do-while loop, jumps in loops.
 Arrays, pointers and structures.

12 hrs

SEQUENCES , FILES AND DICTIONARIES

Sequences, FILE I/O Operations, dictionaries and sets

13 hrs

PROGRAMMING CONCEPTS

10 hrs

Functions , modules and regular expressions

- Text Books
1. Programming in Python 3 (Second Edition) A Complete Introduction to the Python Language by Mark Summerfield
 6. Head First Python – Paul Barry
 7. Gottfried, Byron S., Programming with C, Tata McGraw Hill

ReferenceBooks

5. Problem Solving With C: Jones and Harrow, WILEY INDIA PVT LTD.
6. An Introduction to Programming and Problem Solving With Pascal, G. Michael Schneider Steven W. Weingart David M. Perlman, WILEY INDIA PVT LTD.

SOFTWARES C/PYTHON