MASTER OF COMPUTER APPLICATIONS (MCA)

CHOICE BASED CREDIT SYSTEM (CBCS)

Syllabus
(2017 – 2018)

DEPARTMENT OF COMPUTER SCIENCE, APPLICATIONS & INFORMATION TECHNOLOGY
FACULTY OF ARTS, SCIENCE AND HUMANITIES
SEMESTER-I

17CAP101 INFORMATION TECHNOLOGY 4H - 4C

Instruction Hours / week: L: 4 T: 0 P: 0  Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours

Scope: This course provides basic understanding about the elements of information processing system, programming languages, operating system, computer network and security of computer system.

Objective: The objective of this course is to

- Familiarize the students with PC, MS DOS and MS WINDOWS commands such as file creation, editing and directory creation.
- Learn to use of MS office: MS WORD, use of database and spread sheet, slide creation with PowerPoint.
- Understand about information integrity and security
- Use of a visual programming language such as Visual Basic for various applications.

Unit I
Information concepts and processing: Evolution of information processing, data information language and communication.
Elements of a computer processing system: Hardware - CPU, storage devices and media, VDU, input-output devices, data communication equipment Software- system software, application software.

Unit II
Programming languages: Classification, machine code, assembly language, higher level languages, fourth generation languages.
Operating systems: Concept as resource manager and coordinator of processor, devices and memory. Concept of priorities, protection and parallelism. Command interpreter, Typical commands of DOS/ UNIX/Network, GUI- Windows.

Unit III
Computers and Communication: Single user, multi-user, work station, client server systems, Computer networks, network protocols, LAN, WAN, Internet facilities through WWW, Mosaic, Gopher, html, elements of Java.

Unit IV
Information integrity definition, Ensuring integrity, Computer security: Perverse software, concepts and components of security, Preventive measures and treatment.

Unit V
Range of application: Scientific, business, educational, industrial, national level weather forecasting, remote sensing, planning, multilingual applications.

SUGGESTED READINGS
Scope:
This course provides students with a comprehensive study of problem solving techniques, fundamentals of C and Data structures. It provides programmers with the means of writing efficient, maintainable and portable code applying various data structures.

Objective: Enable the students to
- understand the concept of programming (i.e., computer following a series of instructions)
- exercise various features of C programs (looping, branching, pointers and structures)
- write well structured modular and readable programs with good documentation.
- understand the concept of a program in a high-level language being translated by a compiler into machine language program and then executed.
- know about various data structures such as stacks, Queues, linked lists and trees
- use various data structures in applications such as sorting, searching, string manipulation and list manipulation.

Unit I
Introduction to algorithms, Flow charts, Tracing flow charts, Problem solving methods, Need for computer languages, Reading programs written in C language, C character set, Identifiers and keywords, Data types, Declarations, Expressions, statements and symbolic constants, Input-Output: getchar, putchar, scanf, printf, gets, puts, functions, Pre-processor command: # include, define, ifdef. Preparing and running a complete C program.

Unit II
Operators and expressions: Arithmetic, unary, logical, bit-wise, assignment and conditional operators, Library functions, Control statements: while, do-while, for statements, nested loops. Ifelse, switch, break, continue and goto statements, comma operator.

Unit III
Functions: Defining and accessing: passing arguments, Function prototypes, Recursion, Use of library functions, Storage classes: automatic, external and static variables, Arrays: Defining and processing, Passing to a function, Multi dimensional arrays. Strings, operations on strings.
Unit IV


Unit V

Data Structures: Stacks, queues, lists, trees and their application

SUGGESTED READINGS

Scope: The scope of this course is to teach the basic computer operations and computer architectural concepts with consideration of performance, usability, reliability, and power management.

Objectives: To make students
- understand the architecture of the basic functional units of the computer such as the input output system, memory systems and secondary storage systems.
- acquire basic understanding about the hardware and software interaction.
- acquire knowledge about the Instruction Set and Addressing modes used
- know how the benchmarks that help in evaluating the performance of computer systems.

Unit I

Unit II
Control unit, Data path and control path design, Microprogramming Vs hardwired control, RISC Vs CISC, Pipelining in CPU design: Superscalar processors.

Unit III
Memory system, Storage technologies, Memory array organization, Memory hierarchy, interleaving, cache and virtual memories and architectural aids to implement these.

Unit IV
Input-output devices and characteristics. Input-output processing, bus interface, data transfer techniques, I/O interrupts, channels.

Unit V
Performance evaluation - SPEC marks, Transaction Processing benchmarks.

SUGGESTED READINGS
SEMESTER-I

17CAP104 MATHEMATICAL FOUNDATIONS 4H - 4C

Instruction Hours / week: L: 4 T: 0 P: 0 Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3Hours

Scope: Students learn about Mathematical logics, Relations, Algebraic structures, and enable them apply computers for storing data and solving problems.

Objective: This course provides the students
- basic understanding of mathematical logics, predicate and relational calculus
- knowledge about the algebraic structures
- fundamental idea of applying Graph theory for data representation on computers
- knowledge to apply computers in problem solving

Unit I

Unit II

Unit III

Unit IV

Unit V

SUGGESTED READINGS
SEMESTER-I

17CAP105  INTRODUCTION TO MANAGEMENT FUNCTIONS  4H - 4C

Instruction Hours / week: L: 4 T: 0 P: 0  Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3Hours

Scope: To provide the basic understanding about management functions in various aspects of manufacturing and marketing, financial and human resource management

Objective: Students will be able to know about
- Selection training and appraisal of Human Resource,
- Learn about various Marketing strategies,
- Understand Financial management and analysis
- Apply their knowledge in Manufacturing and Strategic management.

Unit I
HRD : selection, appraisal, training and information systems.

Unit II
Marketing: Understand the concept of marketing mix. These marketing mix elements consist of product policy and design, pricing, choice of marketing intermediaries, methods of physical distribution, use of personal selling, advertising and sales promotion, marketing research, and marketing organization.

Unit III
Finance: Finance function (concept, scope, and its relationship with other functions) : tools of financial analysis (funds and cash flow analysis, ratio, analysis, risk-return trade-off): financial forecasting (profonna income statement and balance sheet, cash flow forecasting under uncertainty, financial planning): estimation and management of working capital (operating cycle concept, inventory, accounts receivables, cash and accounts payables, working capital requirements).

Unit IV

Unit V
Strategy: Firm and its Environment: strategies and resources; industry structure and analysis; evaluation of corporate strategy; strategies for growth and diversification; process of strategic planning.

SUGGESTED READINGS
1. MS DOS Commands – create file, copy, move, and rename files
2. Design a Word document containing an advertisement/ notification column
3. Create a table in word to show student marks and calculate the total and average using formulas
4. Create a word document to show Mail merge
5. Illustrate the use of Review Tools to track changes made in a document
6. Create an Excel sheet illustrate sorting and filtering of data
7. Create MS-Excel sheets showing conditional formatting
8. Create an Excel applications to demonstrate Chart tools
9. Demonstrate Data validation with an example
10. Create a Power point Presentations for your profile
11. Create a Power point Presentations for an Advertisement with Animations
12. Write a Visual Basic program to perform arithmetic operations
13. Write a Visual Basic program to design and code the user Registration form
### SEMESTER-I

#### 17CAP112  
**PROGRAMMING IN C (PRACTICAL)**  
5H - 2C

**Instruction Hours / week:** L: 0 T: 0 P: 5  
**Marks:** Internal: 40  
**End Semester Exam:** 3 Hours

| 1. | SIN and COS Series |
| 2. | Array Operations (Insert, Delete and Display) |
| 3. | Bubble Sorting |
| 4. | Stack |
| 5. | Queue |
| 6. | Implement String functions |
| 7. | Pointers and Arrays, Pointers and function |
| 8. | Recursive function |
| 9. | Dynamic Memory Allocation |
| 10. | Matrix Operations (Addition, Subtraction and Multiplication) |
| 11. | Linked List Operations |
| 12. | Mark sheet preparation using array of structures |
| 13. | Electricity Bill Preparation using Files (Use structures) |
| 14. | Implement TWO Dos commands using Command line arguments |
| 15. | Design an application using VRAM |
| 16. | Display a message every 5 minutes using TSR programming |
1. Create a new company in integrate accounts mode and account with inventory mode
2. Create a primary and sub groups using single or multiple ledger mode
3. Create minimum 10 ledgers using single or multiple ledger, and alter and delete any 2 ledger
4. Create a new company, ledger and record minimum 10 transactions without adjustment
5. Create a new company, ledger and record minimum 10 transactions with any five adjustments and display the relevant results
6. Enter the following voucher
   - Payment vouchers
   - Receipt
   - Purchase
   - Sales
   - Credit note
   - Debit note
   - Journals
   - Memo
   - Optional

7. Prepare trail balance for the company
8. Prepare profit & loss a/c and balance sheet
Scope: This course provides an introduction to computer programming concepts using the C++ Programming language to create structured, modular and re-usable code.

Objectives: The objective of the course is to enable the students to

- Understand the Object Oriented Programming concepts and techniques
- Understand the fundamentals of programming in C++
- Understand and utilize the concepts of procedural abstraction, data abstraction and encapsulation, polymorphism and inheritance with respect to C++ programming language
- Apply their programming knowledge for real time software development

UNIT I


UNIT II

Arrays: Declaring Arrays – Passing Arrays to Functions – Sorting Arrays – Linear Search and Binary Search.


Classes and Data Abstraction : Structure Definitions- Class Scope and Accessing Class Members – Access Functions and Utility Functions- Constructors- Destructors.

UNIT III

Classes : Part II : Introduction- friend functions and friend classes- this pointer- new, delete operator- static class members- Data abstraction and Information hiding- Container Classes and Iterators.

Operator Overloading : Fundamentals and restriction of operator overloading –Overloading Stream Insertion, Stream Extraction, Unary, Binary, ++ and – operators. Inheritance : Base Classes and derived Classes – Protected Members- casting base class pointers to derived class pointers- Using member Functions- public, protected, private inheritance – Direct Base classes and Indirect Base classes- Using constructors and Destructors in Derived Classes.
SEMESTER-II

17CAP201  OBJECT-ORIENTED PROGRAMMING WITH C++  4H - 4C

UNIT IV

Templates: Function, Class templates – Overloading template functions- Templates and Inheritance, friends, static members.

UNIT V
Exception Handling: Basics of C++ Exception handling: try, throw, and catch – Throwing, catching, and rethrowing an exception, Exception specifications.

File Processing: Data Hierarchy- files and streams- Creating, Reading, Updating sequential Access files and Random Access files.

The Preprocessor: #include, # define, #error, #pragma preprocessor directives.

Standard C++ Language Additions: Boolean data type- static_cast, const_cast, reinterpret_cast operator, namespaces.

SUGGESTED READINGS

WEBSITES
2. www.cplusplus.com/
3. www.cppreference.com/
Scope: The scope of this course is to teach students the capabilities and limitations of computer operating systems, process management, processor scheduling, deadlocks, memory management, secondary memory management, file management and I/O systems.

Objectives: To provide students with
- extensive knowledge of principles and modules of operating systems
- fundamental operating system abstractions such as processes, threads, files, semaphores, IPC abstractions, shared memory regions, etc.
- the principles of concurrency and synchronization, and apply them to write correct concurrent programs/software
- basic resource management techniques (scheduling or time management, space management) and principles and how they can be implemented. These also include issues of performance and fairness objectives, avoiding deadlocks, as well as security and protection.

Unit I

Introduction: Evolution of operating systems. Types of operating systems. Different views of the operating system, operating system concepts and structure.

Unit II

Memory Management: Memory management without swapping or paging, swapping, virtual memory, page replacement algorithms, modeling paging algorithms, design issues for paging systems, segmentation.

Unit III

Interprocess Communication and synchronization: The need for interprocess synchronization, mutual exclusion, semaphores, hardware sport for mutual exclusion. queuing implementation of semaphores, classical problems in concurrent programming, critical region and conditional critical region, monitors, messages, deadlocks.
File Systems: File systems, directories, file system implementation, security protection mechanisms.
Unit IV
Input/Output
Principles of I/O Hardware: I/O devices, device controllers, direct memory access.
Principles of I/O Software : Goals, interrupt handlers, device drivers, device independent I/O software. User space I/O software.
Disks: Disk hardware, scheduling algorithms, Error handling, trac-at-a-time caching, RAM Disks.
Clocks: Clock hardware, memory mapped terminals, I/O software. Terminals: Terminal hardware, memory mapped terminals, I/O software.
Processes and Processors in Distributed Systems: Threads, system models, processor allocation, scheduling.

Unit V
Performance Measurement, monitoring and evaluation
Introduction, important trends affecting performance issues, why performance monitoring and evaluation are needed, performance measures, evaluation techniques, bottlenecks and saturation, feedback loops.
Case Studies: MS.DOS, MS WINDOWS, LINUX (UNIX) operating system.

SUGGESTED READINGS


17CAP203 INFORMATION SYSTEMS: ANALYSIS, DESIGN AND IMPLEMENTATION 4H - 4C

Instruction Hours / week: L: 4 T: 0 P: 0  Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3Hours

Scope: Introducing students to the concepts and terms of Analysis and Design documents for a given problem using Unified Modelling Language. Pointing out the importance and function of each UML model throughout the process of object-oriented analysis and design.

Objective: This course provides students
  - an understanding about complex system development
  - the importance of classes and objects and their interrelationships
  - analysis and evaluate complex design structures
  - use Unified Modeling Language to construct various UML models (including use case diagrams, class diagrams, interaction diagrams, state chart diagrams, activity diagrams, and implementation diagrams) using the appropriate notation.

Unit I
Overview of Systems Analysis and Design: Systems Development Life Cycle. Concept and Models: requirements determination, logical design, physical design, test planning implementation planning and performance evaluation; communication, interviewing, presentation skills; group dynamics; risk and feasibility analysis; group-based approaches. JAD, structures walkthroughs, and design and code reviews; prototyping; database design; software quality metrics; application categories software package evaluation and acquisition.

Unit II
Information requirement Analysis: Process modelling with physical and logical data flow diagrams, data modelling with logical entity relationship diagrams; Developing a Proposal: Feasibility study and cost estimation. System Design: Design of input and control, design of output and control, file design/database design, Process design, user interface design; prototyping; software constructions; documentation.

Unit III
Application Development Methodologies and CASE tools: Information engineering, structured systems analysis and design and object oriented methodologies for application development data modeling, process modeling, user interface design and prototyping; use of computer aided software engineering (CASE) tools in the analysis, design and implementation of information systems.

Unit IV
Design and Implementation of 00 platforms: Object oriented analysis and design through object modeling technique, object modeling, dynamic modeling and functional modeling, object oriented design and object oriented programming systems for implementation, object oriented data bases.

Unit V
Managerial Issues in Software Projects: Introduction to software markets; planning of software projects, size and cost estimations; project scheduling; measurement of software quality and productivity; ISO and capability maturity models for organizational growth. The course should be based on lectures, case analysis and laboratory work. Cases should be used to illustrate each major topic in the course.

SUGGESTED READINGS
SEMESTER-II

17CAP204  ACCOUNTING AND MANAGEMENT CONTROL  4H - 4C

Instruction Hours / week: L: 4  T: 0  P: 0  Marks: Internal: 40  External: 60  Total: 100
End Semester Exam: 3 Hours

Scope: The scope of this course includes a variety of aspects of business operations. The main aim of accounting is to help management in its functions of planning, directing, and controlling areas of specialization included within the boundary of management accounting.

Objective: The objective of this course is to acquaint students with the accounting concepts, tools and techniques for managerial decision. The course stresses the relation of accounting to economic activity, organizing information for decision-making, the resource acquisition decision, the uses of cash and non-cash resources, the accounting for selling and manufacturing activities.

Unit I
Basic Accounting and conventions underlying preparation of Financial Statements (balance sheet highlighting accounting equation, profit and loss statement; accounting processes; basic accounts, trial balance and financial statements; issues such as provisions for bad debts tax, dividends, losses such as bad debts, missing information, classification effect, cost of assets, rentals, etc);

Unit II
Income Measurement (revenue; recognition and matching costs and revenues; inventory valuation); Depreciation Accounting; Intangible Assets Accounting; Understanding published annual accounts including funds flow statement.

Unit III
Basic Cost Concepts: (introduction; cost classification; allocation, appointment and absorption; cost centers); Cost Analysis for Managerial Decisions (direct costing, break-even analysis; relevant costs; pricing; pricing-joint costs; make or buy; relevant fixed costs and sunk costs) Cost Analysis for Control (standard costing; variances; material, labour, overhead, sales, and profit); Standard Cost Accounting (budgeting and control; elements of budgeting; control of manufacturing and manufacturing expenses; performances appraisal, evaluation of cost control systems).

Unit IV
Introduction to Management Control Systems; Goals, Strategies, and Key Variables; Performance Measures; Responsibility Centers and Transfer Price; Investment Centers; Reporting Systems; Management by Objectives;

Unit V
Budgeting and Control; Organizational Relationships in Control; Control Dynamics; Top Management and Control; Strategic and Long-Range Planning; Control of Service Organizations; Control of Projects; Control of Non-Profit Organizations; Control of Multinational Companies.
Suggested Readings

Scope: The scope of this course includes probability theory and probability density functions; permutation and combinations and also problems involving principles of inclusion and exclusion. This subject gives the knowledge about Bayes Rule, Total Probability Theorem, Random variables and distributions.

Objective: The objective of this course is to equip the students with the fundamental principle of counting. It also enables them
- To find the permutation when the all objects are distinct or non distinct.
- To find the combination of objects.
- To Understand Connection between Permutations & Combinations.
- How to distinguish between Permutations & Combinations
- How to solve problem involving inclusion and exclusion.

Unit I

Unit II
Distribution and density functions. Marginal and conditional distributions. Stochastic independence.

Unit III

Unit IV

Unit V

Suggested Readings
1. Create a class Date whose data members are Day, month, and Year. Write necessary member functions and perform the following operations using overload operators
   - Increment a date by a day
2. Compares two dates
3. Create a class String that has a character array as a data member and perform the following operations using overloaded operators.
   - ‘+’ To add two strings
   - ‘+=’ To concatenate one string with the other
   - ‘==’ To compare two strings
4. Create a class Computer and derive two classes Client and Server from it. Have the data members of the classes as follows.
   In the main () program, get the data about n clients and servers and print it back in a neat format.
5. Create four classes with the relationship and data members as in shown in the diagram.
   In the main () program, have the facility to
   - Store the details of n inpatients and outpatients
   - Display the details in a neat format
6. Create a class staff that contains the name, designation, and years of experience of a staff member of a college. Using containership, create two more classes Teaching Staff and NonTeachingStaff according to the following specifications. In addition to the properties of the staff class, the Teaching Staff class should contain the highest qualification the staff member possesses and the departments he belongs to. The NonTeachingStaff class needs to contain the properties of Staff only. In the main (), get data about some of the teaching and NonTeachingStaff members of your college and print the details in neat format.
7. Create a class Address as whose data members are Name, Street, City, Pincode and Phone Number of a person. In the main () program, using array of pointers, get addresses of n persons, sort it in alphabetical order of names and display it back in a neat format.
8. Create a class Shape that contains two data members of type double to hold the two dimensions of the shape.
   Derive 3 more classes’ circle, rectangle, And Triangle from the class Shape. Using appropriate member functions, get the values, calculate and print the area of different shapes using dynamic binding.
   Hint:- write 2 member functions in all the derived classes: one to set the data and the other to calculate and display the area.
9. Create two classes British and Metric to store the measurements of distance in the British (feet and inches) and Metric (meters and centimeters) systems respectively. In the main () program, perform the following:
   - Get two measurements: one in British and the other in Metric
   - Ask the user in which system (British or Metric) (s) he wants the output.
   - Add two input measurements and print the result according to the user’s choice.

10. Create a class that copies the content of a text file into another file. Write the program in such a way that the program accepts command line arguments and make the program to execute in a way exactly the copy command in Dos works.

11. Sort an integer and a floating-point array using function template.

12. Create a class template stack that accepts a generic data type as a parameter and performs the basic operations of a stack. Using the above class in the main () program, implement stacks for integer and floating point data types separately and perform the operations on the stack.
1. To write a Linux program to display process deadlock state.
2. To write a program to implement signal handling.
3. Write a Shell program to handle student data base with options given below:
   a) Create data base. b) View Data Base. c) Insert a record d) Delete a record.
   e) Modify a record. f) Result of a particular student. g) Exit.
4. To write a simple Linux program using thread.
5. Deadlock avoidance using Banker's Algorithm.
6. To write a program to display the date & time using TCP Sockets.
7. To write a program to display the date & time using UDP Sockets.
8. Simulation of following CPU scheduling algorithms:
   A. FCFS
   B. SJF (preemptive and non-preemptive)
   C. Priority scheduling (preemptive and non-preemptive)
   D. Round Robin Scheduling
9. To write a Linux program to create a lock file.
10. To write a program to display the user information
The lab sessions will have experiments on the following:

1. **CASE tools**
   Use of diagramming tools for system analysis, such as Turbo analyst, for preparing Data Flow diagrams and E-R diagrams. Use of tools for relational database design such as relational Designer.

2. **Application Development Tools:**
   Use of tools such as Power Builder, Delphi, Magic etc. in developing application software including interactive data-entry screens, transaction processing, report generations, etc.

3. **Management Tools:**
   Use of tools for managing the process of software development such as Source Code Control System (SCCS), Revision Control System (RCS), Make etc.

4. **Program Using Selenium Tool:** [web: http://docs.seleniumhq.org]
   i. Using seleniumhq tool perform single webpage test case
   ii. Using seleniumhq tool perform testing for multiple webpage
Scope: This course aims to provide students with comprehensive and in-depth knowledge of:
- architecture and functioning of database management systems
- to design and built a database system
- integrity constraints, triggers, functional dependencies, and normal forms

Course Objective:
A student who successfully completes this course should, at a minimum, be able to:
- Understand the role and nature of relational database management systems (RDBMS) in today's IT environment.
- Translate written business requirements into conceptual entity-relationship data models.
- Convert conceptual data models into relational database schemas using the SQL Data Definition Language (DDL).
- Query and manipulate databases using the SQL Data Manipulation Language (DML).

UNIT I

UNIT II
Relational Model. Languages & Systems - Relational Data Model & Relational Algebra - Relational Model Concepts - Relational Model Constraints-Relational Algebra - SQL - A Relational Database Language - Date Definition in SQL - View & Queries in SQL - Specifying Constraints & Indexes in SQL - Specifying Constraints & Indexes in SQL - a Relational Database Management Systems - ORACLE/INGRES

UNIT III
UNIT IV
Relational Data Base Design - Function Dependencies & Normalization for Relational Databases - Functional Dependencies - Normal forms based on primary keys - (INF, 2NF, 3NF & BCNF) - Lossless join & Dependency preserving decomposition

UNIT V

SUGGESTED READINGS
**17CAP302  Computer Networks  4H - 4C**

**Instruction Hours / week: L: 4 T: 0 P: 0**

**Marks: Internal: 40 External: 60 Total: 100**

**End Semester Exam: 3Hours**

**Scope:** This course is to provide students the fundamentals of data communications networks, working knowledge of data transmission concepts, understanding the operation of all seven layers of OSI Model and the protocols used in each layer.

**Course Objective:** At the end of the course, students will:
- Become familiar with layered communication architectures (OSI and TCP/IP).
- Understand the client/server model and key application layer protocols.
- Learn sockets programming and how to implement client/server programs.
- Understand the concepts of reliable data transfer and how TCP implements these concepts.
- Know the principles of congestion control and trade-offs in fairness and efficiency.
- Learn the principles of routing and the semantics and syntax of IP.

**UNIT I**

**Introduction to computer network**

Advantages of networks - structure of the communications network - point-to-point and multidrop circuits - data flow and physical circuits - network topologies - topologies and design goals - Hierarchial topology - horizontal topology (Bus) - star topology - ring topology - mesh topology. The telephone network, switched and non-switched options - fundamentals of communications theory - channel speed and bit rate - voice communications and analog waveforms - bandwidth and the frequency spectrum - connecting the analog and digital worlds - digital worlds - digital signals- the modem, asynchronous and synchronous transmission.

Wide area and local networks, connection oriented and connectionless networks, classification of communications protocols, time division multiple access (TDMA), time division multiplexing (TDM), carrier sense (Collision) systems, token passing, peer-to-peer priority systems; priority slot, carrier sense (collision free) systems, token passing (priority) systems.

**UNIT II**

**Layered Protocols and the OSI model**

Goals of Layered Protocols, network design problems" communication between layers, introduction to standard organizations and the OSI model, standards organizations, Layers of OSI, OSI status.

**Local Area Networks**

Way LANs?, Primary attributes of a LAN, Broadband and baseband and base LANs, IEEE LAN standards, relationship of the 802 standards to the ISO/CCITT model., connection options with LANs, LLC and MAC protocol data units, LAN topologies and protocols., token ring (Priority), token bus and IEEE 802.4, metropolitan area networks (MANs), ANSI fiber distributed data interface.
UNIT III

Network Protocols: TCP, UDP, IP, ICMP, SNMP, RMON
TCP/IP: TCP/IP and internetworking, example of TCP/IP operations, related protocols ports and sockets. The IP address structure, major features of IP, IP datagram. Major IP services. IP source routing, value of the transport layer. TCP, Major features of TCP, passive and active operation, the transmission control block (TCP), route discovery protocols, examples of route discovery protocols, application layer protocols.

UNIT IV

Polling/Selection Protocols

Character and bit protocols, binary synchronous control (BSC) HDLC; HOLC options, HDLC frame format, code transparency and synchronization, HDLC transmission process, HDLC subsets, SDLC; Protocol conversion.

Switching and Routing in Networks

Message switching, packet switching, when and when not to use packet switching, packet routing, packet switching support to circuit switching networks.

The X.25 Network and Supporting Protocols


UNIT V


SUGGESTED READINGS

Scope: The course covers both core and advanced Java concepts like Database connectivity, Threads, Exception Handling, Collections, JSP, Servlets, XMLHandling etc. students will also learn various Java frameworks like Spring.

Course Objective: After the completion of the Advance Java Course, students will be able to:
1. Develop the code with various Java data types, conditions and loops.
2. Implement arrays, functions and string handling techniques.
3. Understand object oriented programming through Java using Classes, Objects and various Java concepts like Abstract, Final etc.
4. Implement multi-threading and exception handling.
5. Write a code in JDBC to communicate with Database.
6. Write code with spring framework components.

UNIT I


UNIT II

Files and Streams: Advanced Input/output Streams, Readers and Writers, Character and Byte Streams, PrintWriter, Reading Text, Scanner Class, Reading and Writing Files, Copying a File, Class File, Creating a Sequential, Access File, Reading Data from a Sequential, Access File, Random-Access Files, Creating/Writing/Reading Random-Access Files, New I/O APIs for the Java Platform.

UNIT III

UNIT IV

Introducing the Spring Framework, Spring Framework RunTime & architecture, Inversion of Control (IoC), Dependency Injection, Different Forms of Dependency Injection, Dependency Injection variants, DI classes & Parameter in Spring framework, Bean naming, @Autorwired annotation, The Bean Factory, XML Bean Configuration, Managing the Bean Lifecycle, Basics of Aspect-Oriented Programming (AOP), AOP concepts - Join point, Pointcut, Advice, Types of advice, @AspectJ support

UNIT V

DAO Support and JDBC Framework, Operations with JdbcTemplate, JdbcTemplate Convenience Methods, Basic Queries Using the JdbcTemplate, Batch Updates, Transaction and Resource Management, Global transaction vs. local transaction, Declarative transaction management, XML-based, Annotation-based, Object/Relational Mapping, Basic O/R Mapping, Object Query Languages, Data Access Objects, Setup in a Spring Context, Introduction to Spring MVC, DispatchServlet, Context configuration, Identify the design goals and core concepts of Spring MVC, Spring MVC controllers & Views

SUGGESTED READINGS

7. Rod Johnson, Jurgen Holler & Alef Arendsen. Professional Java Development with the Spring Framework

WEB SITES
1. java.sun.com/docs/books/tutorial/
2. www.en.wikipedia.org/wiki/Java
3. www.java.net/
Scope: On successful completion of this course the learner gains knowledge about the concept of statistical measures, relationship between variables, hypothesis testing, theory of estimation, experimental designs and quality control charts and its applications with exact statistical treatment.

Course Objective: To make the learners to understand the concepts of statistical measures, relationship between variables, hypothesis testing, theory of estimation, experimental designs and quality control charts with computational applications

UNIT I
Statistical Measures: Introduction to descriptive Statistics -basic definitions, frequency distribution, Measure of Central Tendency: Mean Median and Mode. Measure of Dispersion: Absolute and relative measure of dispersion: Range, Mean deviation, Quartile deviation, Standard deviation and corresponding relative measures.

UNIT II
Correlation and Regression: Types of Correlation – Simple and Multiple, Positive and Negative, Linear and Non-Linear, Partial and Total. Methods of calculating correlation coefficient: Scatter diagram, Karl Pearson and Spearmen (Rank) correlation coefficient, Regression: Types, lines and equations, Linear Regression - least square method of solving regression equations, X on Y and Y on X.

UNIT III
Testing of Hypothesis: Introduction to Inferential Statistics: Null and alternative hypothesis, Type I and Type II errors, Standard error, level of significance, acceptance and rejection regions and procedure for testing hypothesis. Large sample test - Z test - tests for means, variances and proportions, Small sample tests based on t, F and Chi- square distributions.

UNIT IV
Estimation and Design of Experiment: Point Estimation - characteristics of estimation - interval estimation -interval estimates of mean, standard deviation and proportion. Design of Experiments: Completely Randomized Design (CRD) , Randomized Block Design (RBD) and Latin Square design (LSD) Models

UNIT V

SUGGESTED READINGS
Scope: The scope of the course includes Decision Support System, which deliver information and knowledge to different people and levels in the firm—operational employees, middle managers, and senior executives.

Course Objective: The objective of the course gives knowledge for the students about decision support system (DSS) which is a computer program application that analyzes business data and presents it so that they can make business decisions more easily. It helps students to distinguish it from an "operational application" that collects the data in the course of normal business operation.

UNIT I
Introduction to the concept of Decision Support System: Components of DSS: Dialogue Management;

UNIT II
Data Management and Model Management for DSS; Examples of different type of DSS; Systems Analysis and Design for DSS;

UNIT III
Models in the context of DSS; Algorithms and Heuristics; DSS Applications in different functions

UNIT VI
Design of interfaces in DSS; An overview of DSS generators; Group Decision in Support Systems (GDSS) and Decision Conferencing.

UNIT V

SUGGESTED READINGS
List of Practical Programs

Tool: MySQL

Some sample applications which may be programmed, are given below

I  Create following tables with appropriate constraints.

1. Write SQL statement for following queries.
   a. Age of employees.
   b. Employees whose take home salary is in the range Rs10,000 and 15,000
   c. Employees whose has put 10 years of service.
   d. Employees working under the department head RAGURAMAN
   e. Senior and youngest employee in each department.
   f. Employees who retires after one year.

2. Write cursors
   a. To raise the basic salaries by 25 percent
   b. To retain male employee in the existing employee table and to move female employees to a new table.

3. Create following views
   a. For male employees of age more than 45 years.
   b. For operator with the CHECK OPTION.
   c. For Female employees working under the department head RAJARAMAN.
   d. For male employees with the CHECK OPTION.

4. Create a trigger which fires when one tries
   a. To update records in employee table on Sundays.
   b. To insert records in salary table with basic >30000

5. Generate a pay slip where 40 percent, 20 percent and 15 percent of basic is given as HRA, TA and DA for the employees.

6. Write SQL statement selects all employees and all basic salaries using FULL OUTER JOIN

II  Create following tables with appropriate constraints.

1. Take a stock report and list of books in circulation.

2. Write SQL statements for following queries.
   a. Authors of C and C++ books.
   b. List of books issued for UG student
   c. Title of books in computer science department.
   d. Titles of book by the author ‘DIETEL’.
   e. Publishers of C and C++ books.
   f. Number of book issued to each student
   g. Publisher details of Commerce Department.
III Write a cursor
   a. To move books costing more than Rs.1500 into a separate table, Reference table.
   b. To move damaged books into a separate table damaged table.

Some sample PL/SQL Programs

1. Write a recursive program for finding the factorial of a given number.
2. Write a recursive program for finding the first n Fibonacci number.
3. Write a PL/SQL program for multiplication tables 3, 4, 5 and 6.
4. Write a recursive program for finding the reverse of a given number.
5. Write SQL queries to illustrate the string functions mathematical functions.
6. Write a program for finding the reverse of a given string.
7. Write a program using triggers, joins and SQL developer programs
1. Write a networking program in Java to implement a TCP server that provides services for a TCP Client.

2. Write a networking program to implement socket programming using User datagram Protocol in Java.

3. Implement an FTP server using socket programming.

4. Implement a chat server using socket programming.

5. Implement an ECHO server using socket programming.


7. Implement Ping server and Ping client using socket programming.

8. Using UDP to transfer a text file from one host to another.


10. Simulate simple Web Browser.

11. Write a Java program to check whether the given DNS is found in the internet or not.

12. Write a network program using HTTP to print the document for the given URL.
1. Write a java Thread java program by Extending thread Class
2. Write a java program to get file list from a folder filtered by Extensions
3. Write a java program to sort Linked List Using Comparator
4. Write a java program to read all elements in Array List by using Iterator
5. Write a java program to copy Map content to another Hash Table
6. Write a java program to compare two sets and retain elements which are same on both sides
7. Write a java program to get spring Application Context Object Reference
8. Write a java program to load multiple java based configuration in the spring using IMPORT @ Annotation
9. Write a java spring AOC Advice using PointCuts concept – to check the name match
10. Write a java program to update customer Information Using JDBC Connection
11. Write a java program to Execute select Query Using JDBC Connection
12. Write a java program to get IP Address of a Host in Java
SEMESTER-IV

17CAP401

J2EE

4H -4C

Instruction Hours / week: L: 4 T: 0 P: 0 C:4

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Scope: This is designed to rapidly learn Java web programming with J2EE. To understand object-oriented programming with J2EE and learn how to write increasingly sophisticated J2EE programs and to get started fast in J2EE programming.

Objective:

- Understand J2EE as an architecture and platform for building and deploying web-based, n-tier, transactional, component-based enterprise applications
- Understand the fundamental concepts of XML and related technologies
- Acquire knowledge on how XML is currently being used in various application areas
- Know how to parse and transform XML documents via tools and through programming APIs
- Understand the EJB architecture and have a good grasp on when to use and how to use various EJB bean types and acquire relevant Java programming experience

UNIT I


UNIT II


UNIT III

Java Servlets: Benefits – Anatomy – Reading Data from Client – Reading HTTP Request Headers – Sending Data to client – Working with Cookies.

UNIT IV


UNIT V

Java Server Pages – Java Remote Method Invocation.
SUGGESTED READINGS


WEB SITES

1. java.sun.com/javaee/
2. java.sun.com/j2ee/1.4/docs/tutorial/doc/
3. www.j2eebrain.com/
Scope: Mobile computing is a technology that allows transmission of data, via a computer, without having to be connected to a fixed physical link. This technology gives the ability to send and receive data across these cellular networks. Mobile data communication has become a very important and rapidly evolving technology as it allows users to transmit data from remote locations to other remote or fixed locations.

Objective: The aim of the course is to introduce an advanced element of learning in the field of wireless communication and the concepts of wireless devices and mobile computing. It also introduces wireless communication and networking principles, that support connectivity to cellular networks, wireless internet and sensor devices. The course helps to understand the use of transaction and e-commerce principles over devices to support mobile business concepts and to learn the basic concepts, aware of the GSM, SMS, GPRS Architecture. It also appreciates to understand the mobile operating system development environment such as android.

UNIT I

UNIT II

UNIT III

UNIT IV
UNIT V
Android: Getting to know Android - Android development environment - Android development environment for real applications - start up code, M J Android applications- debugging Android applications

SUGGESTED READINGS


WEBSITES:
1. en.wikipedia.org/wiki/Mobile_computing
2. www.cse.iitk.ac.in/users/rkg/Talks/mobile_main.pdf
3. www.tutorialspoint.com/android/
4. pl.cs.jhu.edu/oose/resources/android/Android-Tutorial.pdf
Scope: This course makes to understand the management principles and organizational concepts in
- Conduct of an organization. It makes to plan and execute the given activity with less conflict and
- Gaining maximum values to the organization.
- It is an essential interdisciplinary approach to study human behaviour at work.

Objectives: To make the students familiar with basic concepts of management & human behavior in an Organizational context and application of these concepts to managerial problems. This course tries to integrate the relevant knowledge drawn from related disciplines like psychology, sociology and anthropology to make them applicable for studying and analysing organizational behaviour.

Unit I
Management overview: – evolution of management - functions - characteristics and
Importance of management - management vs administration - role of manager - planning: Meaning -
steps in planning - objectives - management by objectives.

Unit II
Decision making: meaning, process. Organization – departmentalization - manpower planning -
Process of recruitment and selection, placement and orientation - directing – motivation –
Communication. Control – need, process, steps in control.

Unit III
Organization behavior: meaning and definition - fundamental concepts of ob – contributing
Disciplines to the ob field - learning – theories of learning - personality – determinants of Personality
- theories of personality - psycho-analytical, social learning, job-fit, and trait Theories.

Unit IV
Perception – factors influencing perception – attribution theory – frequently used shortcuts in
Judging others - perceptual process - organization errors of perception - attitudes – types of Attitudes
- functions of attitudes. Values – types of values - value across cultures – groups -Stages of group
development - group norms - group cohesiveness.

Unit V
Conflict – meaning – functional vs. Dysfunctional conflict - levels of conflict – conflict
Organizational change - forces for change - resistance to change- overcoming resistance To change.

6. Gilbert,(2008),”management today principles and practice”, tata mcgraw hill,
7. Fred luthans(1995),”organizational behavior”, 7th edition, tata mcgraw hill,

**Journals/ additional readings:**
1. Effective executive – magazine
2. All management journals
3. Vikalpa the journal for decision makers
4. Harvard business review
5. Human resource management review
6. Training and development journal
7. Personnel today
8. Academy of management journal
9. Academy of management review
10. The iup journal of organization behavior
### Scope:
The scope of this course is to teach students how to perform day-to-day database administrative tasks
- To keep database operational
- Basic trouble shooting
- Monitoring performance of tasks

### Objective:
- Install and configure database
- Create users and assign roles
- Optimize schemas, tables, indexes and views
- Manage database services and clients
- Will be able to move the data from one database to another database.
- Take backup and perform recovery.

#### UNIT I

#### UNIT II

#### UNIT III
UNIT IV

UNIT-V
Creating Table spaces in a VLDB Environment: Bigfile Table space Basics – Creating and Modifying Bigfile Tablespace – Bigfile Tablespace ROWID format – DBMS_ROWID and Bigfile Tablespaces.- Advanced Oracle Table Types – Using Bitmap Indexes – Oracle Data Pump
Remote queries – Remote Data Manipulation: Two Phase Commit – Managing Distributed Data – Managing Distributed Transactions – Monitoring and Tuning Distributed Database

SUGGESTED READINGS
2. Saikat Basak.(2010), Oracle DBA Concise Handbook ,Ensel Software

WEBSITES:
- www.oracle.com/technology/software/products/database/oracle10g/index.html
- www.oracle-base.com/articles/10g/
- www.adp-gmbh.ch/ora/misc/10g.html
Scope: To make student understand the goals, issues, technologies, algorithms, protocols and design criteria used in cryptography and data security and solution.

Objective:
- To teach fundamental aspects of security in a modern networked environment with the focus on system design aspects and cryptography in the specific context of network.
- To build protection mechanisms in order to secure computer networks.
- Write coding to encrypt “Plain Text” into “Cipher Text” and vice versa, using different encryption algorithms.
- The ability to choose a suitable ciphering algorithm according to the required security level.
- Build cryptosystems by applying encryption algorithms,
- Build secure authentication systems by use of message authentication techniques.

UNIT I

UNIT II

UNIT III

UNIT IV
UNIT V

SUGGESTED READINGS

WEB SITES
1. williamstallings.com/Crypto3e.html
2. u.cs.biu.ac.il/~herzbea/book.html
3. www.flipkart.com/search-books/cryptography+and+network+security+William+stallings+ebook
Scope: Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs.

Objective:
- To perform test each time they are run, thereby eliminating human error.
- To test how the software reacts under repeated execution of the same operations.
- To program sophisticated tests which bring out hidden information from the application.
- To reuse tests on different versions of an application, even if the user interfaces changes.

UNIT I
Introduction: Purpose of Testing – Dichotomies - Model for Testing – consequences of bugs-
Taxonomy for bugs

UNIT II
Flow/Graphs and Path Testing : Path testing basics-predicates, path predicates and Achievable
paths – Path sensitizing– path instrumentation-implementation and application of path testing

UNIT III
Transaction flow testing: – Transaction flow - Transaction flow testing techniques –
implementation comments – testability tips -Data flow testing basics- Data flow testing
strategies.

UNIT IV
Domains and paths-Nice Domains and Ugly Domains-Domain testing-Domains and interface
testing-Domains and testability-Metrics-Linguistic metrics-structural metrics-Hybrid metrics-
metrics implementations-Testability tips

UNIT V
Analysis – MITs Ranking Criteria - Using Risk Ranking in Forensics –Test estimation process-
MIT totals worksheet-Sizing worksheet
SUGGESTED READINGS

WEB SITES
1. http://my.safaribooksonline.com
2. http://www2.sas.com
4. www.cs.cmu.edu
5. www.softwaretesting management.con
Scope:
XML is a universal standard for information markup that is revolutionizing the way of handling data. This course will help students to learn how to create websites using XML, understand the essentials of the XML standards and appreciate some of the performance characteristics, gain some practical experience and insight of using XML.

Objective: To help students to
- Be aware of a range of XML tools (Many of them are free).
- Know how to set out an XML document
- Define custom markup language
- Understand the purpose of using DTDs and Schemas to validate XML
- Use XSLT to write a style sheet for XML document to produce multiple output
- Combine XML with existing web technologies.

UNIT I

UNIT II

UNIT III

UNIT IV
XML Path Language: Nodes – Location Paths; XSLT: Templates - Creating Elements and attributes – Iteration and Sorting – Conditional Processing – Copying Nodes – Combining style sheets – variables. XLink, XPointer, XInclude and XBase.
UNIT V

SUGGESTED READINGS

WEBSITES
1. en.wikipedia.org/wiki/XML
2. www.w3.org/XML/
3. www.w3schools.com/xml/default.asp
Scope: The scope of managerial economics refers to its area of study. Managerial economics has its roots in economic theory. The empirical nature of managerial economics makes its scope wider. Managerial economics provides management with strategic planning tools that can be used to get a clear perspective of the way the business world works and what can be done to maintain profitability in an ever changing environment.

Objective: The purpose of this course is to provide students with a basic understanding of the economic theory and analytical tools that can be used in decision making problems. Students who successfully complete the course will have a good understanding of economic concepts and tools that have direct managerial applications.

The course will sharpen their analytical skills through integrating their knowledge of the economic theory with decision making techniques. Students will learn to use economic models to isolate the relevant elements of a managerial problem, identify their relationships, and formulate them into a managerial model to which decision making tools can be applied. In addition, the course will provide a basic introduction to econometric analysis and its role in managerial decision making.

UNIT I

UNIT II

UNIT III

UNIT IV

UNIT V
SUGGESTED READINGS

WEB SITES
http://www.pondiuni.edu.in/sites/default/files/Managerial%20Economics.pdf
http://www.bput.ac.in/lecture_notes/ME%20Managerial%20Economics.pdf
Scope: The Scope of this course is to expose the students to the architecture, design, and implementation of massive-scale data systems. The course also discusses foundational concepts of distributed database theory including design and architecture, security, integrity, query processing and optimization, transaction management, concurrency control, and fault tolerance.

Objectives: To impart necessary skills to students
- To design good performing distributed database schemas.
- To create optimized query execution plan.
- To efficiently distribute and manage the data.
- To manage distributed access control
- To know how to make secure the databases.

UNIT - I
Database concepts: Data Models- Database Operations- Database Management-DB Clients, Servers, and Environments. DBE Architecture: Services- Components and Subsystems- Sites - Expected Services-Expected Subsystems- Typical DBMS Services– DBE Taxonomy: COS Distribution and Deployment- COS Closeness or Openness-Schema and Data Visibility-Schema and Data Control.

UNIT - II

UNIT - III

UNIT - IV

UNIT - V
SUGGESTED READINGS


WEBSITES

1. en.wikipedia.org/wiki/Distributed_computing
Scope: The goal is to understand the basics of the layered TCP/IP stack architecture for IPv4 and the key protocols in the TCP/IP suite including Internet Protocol version 4 (IPv4), Internet Control Message Protocol (ICMP), Transmission Control Protocol (TCP), and User Datagram Protocol (UDP). Knowledge of IPv6 address auto configuration works for stateless address auto configuration.

Objective: To educate students

- Learn about IPv4 forwarding and routing.
- Learn about host name resolution and the Domain Name System (DNS).
- Learn about IPv6 addresses.
- know Architectural Overview of the TCP/IP Protocol Suite

UNIT I

UNIT II

UNIT III

UNIT IV

UNIT V
SUGGESTED READINGS

WEB SITES
1. en.wikipedia.org/wiki/Internet_protocol_suite
3. www.yale.edu/pelt/COMM/TCPIP.HTM
4. www.w3schools.com/tcpip/default.asp


**Scope**: Introducing students to the concepts and terms used in the object-oriented approach to systems analysis and design. Highlighting the importance of Object Oriented Analysis and Design concepts and apply them to solve problems. Showing how we apply the process of Object Oriented Analysis and Design documents for a given problem using Unified Modelling Language. Pointing out the importance and function of each UML model throughout the process of object-oriented analysis and design.

**Objective**:
- Use object-oriented technologies
- Use Unified Modeling Language 2.2
- Perform object-oriented analysis and design
- Explain how the Unified Modeling Language (UML) represents an object-oriented system using a number of modeling views.
- Construct various UML models (including use case diagrams, class diagrams, interaction diagrams, state chart diagrams, activity diagrams, and implementation diagrams) using the appropriate notation.

**UNIT I**
Complexity: The structure of complex systems—the inherent complexity of software-The five attributes of a complex system-Organized and Disorganized complexity—Bringing order to Chaos-On Designing complex systems. The Object model: The evolution of the object model—Foundations of the object model-Elements of the object model—Applying the object model.

**UNIT II**
Classes and Objects: The nature of an object—Relationship among objects-The nature of a class—Relationships among classes-The interplay of classes and objects—On building quality classes and objects.

**UNIT III**

**UNIT IV**

**UNIT V**
SUGGESTED READINGS


WEB SITES
1. uml-tutorials.trireme.com/
**Scope:** This course will give students an overview of the web services architectures, and then learn the standard APIs for SOAP messaging and WSDL-driven, component-based service development.

**Objective:**
- Be able to describe the interoperable web services architecture, including the roles of SOAP and WSDL.
- Use lower-level SOAP and XML APIs for services and/or clients.
- Build and Host Web Services.

**UNIT I**
Introduction: What are Web Services – Importance of web services – Web services and enterprises; XML Fundamentals:: XML Documents - Namespaces – Schema – Processing XML.

**UNIT II**

**UNIT III**

**UNIT IV**
UNIT V


SUGGESTED READINGS


WEB SITES

1. www.w3schools.com/webservices/default.asp
2. en.wikipedia.org/wiki/Web_service
3. www.webservices.org/
**17CAP405B**

**CORPORATE PLANNING**

**Semester-IV**

Instruction Hours / week: L: 4 T: 0 P: 0 C :43

**Marks:** Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

**Scope:** This course is an introduction to corporate planning with tools and techniques available to managers toward achieving organizational effectiveness. In this course, you will be exposed to the basic tools and techniques available to managers for planning and decision making. The course will also cover a number of topics including the functions of planning, necessity for planning, importance of planning, problems solving in organizational planning issues such as capital budgeting, new product launches and acquisition.

**Objective:** On successful completion of the course, students will be able to:
- Explain the relationship between planning and organizational performance.
- Define corporate planning.
- Explain why do corporate plan fail and what can be done to ensure its success. Explain when should a strategic plan be made.
- Define objectives and why are they considered the foundation of planning
- Identify an organisation’s stated objectives.

**UNIT I**


**UNIT II**

Social responsibilities: Scope, contents, cooperation and society, consumers, corporation and democracy, community and government.

**UNIT III**

Social responsibility versus profitability and productivity. growth. Professionalism as a means of social behaviour.

**UNIT IV**


**UNIT V**


**SUGGESTED READINGS**

1. Create a sign in form in servlets.
2. Write a servlet Program to lock a server.
3. Write a servlet program that returns list of information in table format.
4. Design a counter that counts number of times user has visited the site in current browsing session.
5. Write a program to create JDBC connectivity with Oracle Database.
6. Build a JAVA Bean for opening an applet from JAR file.
7. Write a program to create RMI client and server to invoke remove method of RMI server.
8. Design a counter in JAVA BEAN.
9. Write a program to stream contents of a file using JSP.
10. Write a program to insert a menu applet into JSP page.
1. Write a program to build your first Android Application “Hello World” with common activity.
2. Write a program which will implement Sub menu in android application.
3. Write a program which will implement Context menu (Floating List of Menu Items) in android application.
4. Write a program to displays the use of Relative Layout Views with different attributes.
5. Write a program to displays the use of Linear Layout Views with different attributes.
6. Write a program to implement a Custom Button and handle the display message on button press.
7. Write a program to implement the List View in your android application.
8. Write a program to implement between animations and rotate the text in your android application.
9. Write a sample program to create a progress bar for your android applications.
10. Write a program to show how to use Date picker control of ADK in your android applications.
11. Write a program which enables you to draw an image using bitmap class object.
12. Write a program which allows you to get image from web and displayed them using the Image View.
1. Demo for Globalization Support
2. Setup Listener Security
3. Configuring Recovery Manager
4. Write a program Using Recovery Manager
5. Write a program for Managing Diagnostic Sources
6. Implement Database Recovery
7. Demo for Flashback Database
8. Implement User Error Recovery
9. Write a program for Dealing with Corruption
10. Show the demo for Automated Management
11. Creating a database and do the manipulation.
12. Managing index tables
1. Write a program to convert your college name from plain text to cipher text using
   Transposition cipher method of encryption.
2. Write a program to convert your name from plain text to cipher text using the One Time
   Pads method of encryption.
3. Write a program to encrypt a paragraph using the Data Encryption Standard Algorithm.
4. Write a program to encrypt your biodata using the Advanced Encryption Standard
   Algorithm.
5. Write a program to decrypt the “Network Security” theory syllabus using the RSA
   Algorithm.
6. Write a program that takes a binary file as input and performs bit stuffing and Cyclic
   Redundancy Check Computation.
7. Write a program to Simulate the working of Sliding-Window protocol.
8. Write a program to find the shortest path in a network using Dijkstra's Algorithm.
9. Write a program for the following chat application:
   One to One : Open a Socket connection and display what is written by one to another.
   Many to Many : Each Client Opens a Socket connection to the client server and writes to
   the socket. Whatever is written by one can be seen by all. Implement symmetric key
   cryptography.
10. Write a program to implement International Data Encryption Algorithm (IDEA).
SOFTWARE TESTING

1. Create a VB form with the following fields and create the database also for them. Insert 3 records. Using Win Runner tool record the above 3 transaction and test them and produce the Report. (Blackbox Testing).

2. Create a VB form and then add login dialog form. Using Win Runner tool check the Username and Password and produce the Report. (Security testing).

3. Create a VB form with the following fields and check the calculation is correct or not by using the test toll Win Runner. (Functional Testing) Fields – Name, Designation, Department, Basic, HRA, DA, PF and netsal.


5. Write a C program for Boundary Testing.

6. Write a C program for Loop Testing.

7. Write a C program for Integration Testing.

8. Write a C program for Interface Testing.

9. Write a C program for Unit testing.

10. Write a program to do JUnit test for comparing two list of user defined objects.

Software Quality Assurance

1. To develop a banking application, perform the requirement analysis and give a quality status report.

2. Perform the system testing to develop a electricity application and give a quality status report.

3. Perform the report design and give a quality status report.

4. Develop a library management system and give a quality status report.

5. Develop a hospital management system and a quality status report and give a comparison performance report in Linux and windows operating system.
1. Create a menu in XML.
2. Create a demo for XSLT.
3. Display XML information in Tree structure format.
4. Integrate XML in Web Applications
5. Write a program to navigate the records in the file.
6. Write a database access with XML.
7. Write a program to extract all the information about circles which are elements in a given XML document using DOM parser.
8. Write a program to extract all the information about circles which are elements in a given XML document using SAX parser.
9. Write a program to show the function of CDATA.
10. Write a program to generate XML file on the server.
11. Write a program to generate XML file from the Database
12. Write a program to load a text file into a div element with XML HTTP.
13. Write a program to create a simple application for web services.
1. Demonstrate the creation of output files for Xgraph.
2. Illustrate NS2 script to send data between two nodes.
3. Create a simple simulation topology to generate TCP and UDP traffic
4. Simulate a NS2 scenario to handle link failures.
5. Write a NS2 script to handle Multicast traffic.
6. Demonstrate OSPF routing.
7. Create a simulation topology and analyze QOS parameters.
8. Demonstrate the creation of wireless topology.
SEMESTER-V

17CAP501 PHP5/MYSQL 4H - 4C
Instruction Hours / week: L: 4 T: 0 P: 0 Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3Hours

Scope: PHP course is designed to learn how to do client-side programming, which will run on either a LAMP or a Windows web server.

Objective: To help students to
- Get hands-on experience in scripting, debugging, testing.
- Establish a working environment for PHP web page development
- Use variables, constants, and environment variables in a PHP program
- Learn to create dynamic interactive pages with PHP.
- Learn to manipulate files with PHP.
- Understand how MySQL works.
- Learn to use SQL to output reports with MySQL

UNIT I

HTML Primer: The HTML Document type definition- The Form and Input Elements; Accessing PHP and HTTP Data: Predefined Variables- Variables in HTTP Request and Response- Super Global Arrays; Links; Query Strings; HTML(Web) Forms; HTML Form Elements-HTML Form Fields(Controls) and PHP; The Concept Of State: State Maintenance-Native Sessions in PHP.


UNIT II
Testing and Debugging: Values that break your code- Basic error types; Debugging PHP Script: Understanding PHP error Massages- Syntax Errors- Logic Errors-Runtime Errors; Debugging and Handling Errors in PHP5: Preventing the display of private information-

Roll your-Own Debugging tools; Form Validation: Using the Exit statement- string validation and regular expressions- validating data entry- using reg exps to check file path parameters; Handling Errors: Gracefully- Configuring PHP for error handling- Try/Catch-New in PHP5.
Development planning: Formal software Development processes – optimizing your code- Using Coding standard; Writing user-defined functions in PHP: The Structure of Functions- Switching Functions – How Values Get Inside functions; Scope of variables: Global and Local Variables-Creating Static Function Variable-Nesting-Recursion-The Include and Require Statements-Things to be careful about with include and require.

UNIT III
Files and Directories: Files and Directory Handling- Working with Files- Opening and Closing files-Getting Information about a file-reading and writing to files-Reading and writing characters in files-Reading Entire files-Random Access to file data-Getting Information on Files-Ownership and permissions; Working with files you own: Splitting the Name and path from a file-copying ,renaming and deleting files; Working with Directories: other Directory Functions –Traversing a directory hierarchy-creating a directory navigator-Building a Text Editor-Uploading Files.

Classes- Objects: Creating class- Adding a Method- Adding a Property- Protecting Access to Member Variables- Using _get and _set- Initializing objects- Destroying Objects- Inheritance-Overriding Methods- Interfaces- Encapsulation

UNIT IV
The SQL Framework- Managing databases-Creating & Managing tables- Managing indexes; Inserting & Updating data in a MYSQL database-Deleting & Retrieving data from a MySQL database; SELECT statement- Optional clauses of a SELECT statement; Creating MySQL Expressions-using operators in expressions-Comparing and Converting Data; Managing different types of data: String functions-Numeric function- Date/Time functions- Summarizing date-Summary functions

Performing System Operations: Encryption functions- System related Functions- Query and Insert Functions; Accessing data from Multiple tables: Creating joins in your SQL statement-Creating subqueries in your SQL statements; Creating Unions that join SELECT statements.

Exporting, Copying and importing data; Managing transactions: Introducing transactions-Performing a transaction- Setting the auto commit mode and transaction isolation level- Locking Nontransactional tables

UNIT V
Connecting to MySQL from a PHP application- Inserting and updating records in table- Deleting and retrieving data from table- Creating a user Registration Script. Structure of an E-Mail Message-sending E-mail with PHP- Working with Raster Images- Manipulating Raster Images-Using Text in Images
SUGGESTED READINGS

1. Dave W. Mercer, Allan Kent, Steven D. Nowicki, Davd Mercer, Dan Squie, Wankyu Choi. (2009), Beginning PHP5. Wiley India (P) Ltd, New Delhi

WEB SITES

1. www.php.net/
2. en.wikipedia.org/wiki/PHP
3. www.w3schools.com/PHP/DEfaULT.asp
Scope: This course provides functional understanding of the components of the .NET Framework and the common language runtime (CLR) and know how COM components. Also provides better knowledge of Architecture and Basic Concepts required for building database application with ADO.NET Programming. It also provides an platform to apply all the window controls.

Objective:
- Become a good .NET programmer.
- Know how COM components and the .NET Framework interoperate with each other.
- Identify and use the classes and namespaces in the .NET Framework class library.
- Build WEB Applications using Microsoft ASP.NET programming.

UNIT I

UNIT II
Basic Windows Controls: Textbox Control- ListBox, CheckedListBox-Scrollbar and TrackBar Controls-More Windows Control-The common Dialog Controls-The Rich TextBox Control - Handling Strings, characters and Dates. The TreeView and ListView Controls: Examining the Advanced Controls-The TreeView Control-The ListView Control

UNIT III
The Multiple Document Interface-Databases: Architecture and Basic Concepts-Building Database Application with ADO.NET-Programming with ADO.NET

UNIT IV
Goal of ASP.NET –ASP.NET Web Server Control-Validation Server Controls-Themes and Skins -Content Page Holder

UNIT V
Data Binding in ASP.Net: Data source Controls – Configuring data source control caching – storing connection information-Using Bound list controls with Data Source Controls – Other Databound Controls-Data Management with ADO.Net
SUGGESTED READINGS

WEB SITES
1. www.microsoft.com/NET/
3. www.w3schools.com/ngws/default.asp
4. www.vbtutot.com
Semester-V

17CAP503  OPTIMIZATION TECHNIQUES  4H - 4C

Instruction Hours / week: L: 4 T: 0 P: 0  Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3Hours

Scope
Optimization Techniques is used to design construction and maintenance of engineering systems involved in decision making both at the managerial and technological level.

Objective
- To understand the need and origin of the optimization methods
- To get a broad picture of the various applications of optimization methods used in engineering
- The process of finding the conditions that give the minimum or maximum value of the function where the function represents the effort required or the desired benefits

Unit I
Optimization Techniques Linear Programming: Graphical method for two dimensional problems - Central problem of linear programming various definitions - statements of basic theorems and properties - Phase I and Phase II of the simplex method - revised simplex method - primal and dual - dual simplex method - sensitivity analysis transportation problem and its solution - assignment problem and its solution by Hungarian method.

Unit II
Integer Programming: Gomory cutting plane methods - Branch and Bound method. Queueing Theory: Characteristics of queuing systems - steady state M/M/I, M/M/1/K and M/M/C queueing models.

Unit III
Replacement Theory: replacement of items that deteriorate - Replacement of items that fail Group replacement and individual replacement.

Unit IV
Inventory Theory: Costs involved in inventory problems - single item deterministic models - economic lot size models without shortages and with shortages having production rate infinite and finite.

Unit V
PERT and CPM: Arrow networks - time estimates - earliest expected time, latest allowable occurrence time and slack - critical path - probability of meeting scheduled date of completion of project calculations on CPM network - various floats for activities - critical path - updating project - operation time cost trade off curve - project time cost trade off curve - selection of schedule based on cost analysis.

(Remarks: No mathematical derivations included).
SUGGESTED READINGS

Scope: Scope is to teach the techniques for preprocessing data before mining, and the concepts related to data warehousing, on-line analytical processing (OLAP), and data generalization. It presents methods for mining frequent patterns, associations, and correlations.

Objective: To help the students

- To Possess knowledge of the concepts and terminology associated with database systems, statistics, and machine learning
- To develop skills of using recent data mining software for solving practical problems.
- To gain experience of doing independent study and research.

UNIT I
Introduction to Data Mining: Motivation and importance, Data Mining, Relational Databases, Data Warehouses, Transactional Databases, Advanced Database Systems and Advanced Database Applications, Data Mining Functionalities, Pattern Classification of Data Mining Systems, Major issues in Data Mining. Pre-process the Data- Data Cleaning, Data Integration and Transformation.

UNIT II

UNIT III
Mining Association rule in large Databases Association Rule Mining, Mining Single - Dimensional Boolean Association Rules from Transactional Databases, Mining Multilevel Association Rules from Transaction Databases, Mining Multidimensional Association Rules from Relational Databases and Dataware houses, From Association Mining to Correlation Analysis, Constraint-Based Association Mining.

UNIT IV
Mining Complex Types of Data: Mining Spatial Databases – Multimedia Databases – Time-series and Sequence Data – Text Databases – Web Data Mining – Search Engines.
UNIT V
Data Warehouse and OLAP Technology for Data Mining. What is a Data Warehouse? Multi-Dimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Development of Data Cube Technology, Data Warehousing to Data Mining Data Preprocessing 


SUGGESTED READINGS

1. Jiawei Han and Micheline Kamber. (2011), Data Mining Concepts and Techniques, 3rd Edition, Elsivier,India (Unit I, III, IV, V)
2. G.K.Gupta (2006), Introduction to Data Mining with Case Studies, Prentice Hall India, New Delhi (Unit – IV)

WEB SITES:
2. www.anderson.ucla.edu/faculty/jason.frand/teacher/technologies/palace/datamining.html
Scope: Scope of this course is to teach the concepts and techniques of network architecture management. Know how to Remote Network Monitoring in TCP/IP Networks and general concepts and architecture behind standards based network management

Objective:
- Understand concepts and terminology associated with SNMP and TMN
- Decide routing protocol for complex network.
- Gain knowledge the internal architecture of routers
- Understand the fundamentals and requirements for packet routing in computer communication network.

UNIT I

UNIT-II

UNIT-III
UNIT-IV

UNIT-V

SUGGESTED READINGS
Scope:
To deliver students skills for planning, implementing a software projects and member of a team to develop and deliver.

Objective: To help students to

- Design, select and apply the most appropriate software engineering process for a given project, plan for a software project, identify its scope and risks, and estimate its cost and time.
- Implement software.
- Analyze, specify and document software requirements for a software system.
- Express and understand the importance of negotiation, effective work habits, leadership, and good communication with stakeholders, in written and oral forms, in a typical software development environment.

UNIT I
Introduction-Software Project Management -Project evaluation and programme Management- An overview of Project planning- Stepwise planning-Selection of an appropriate project Approach.

UNIT II
Software effort estimation: Problems with over- and underestimates-Software effort estimation Techniques - Estimating by analogy -Albrecht function point analysis -Function points Mark II – COSMIC full function points - COCOMO 13: a parametric productivity model. Activity planning: The objectives of activity Planning-Project schedules - Projects and activities - Sequencing and scheduling activities - Network planning models - Formulating a network model - Adding the time dimension - The forward pass - The backward pass - Identifying the critical path.

UNIT III
UNIT IV
Monitoring and control: Creating the framework-Collecting the data- Visualizing progress- Cost monitoring -Earned value analysis-Prioritizing monitoring - Getting the project back to target - Change control. Managing people in software environments: Understanding behavior - Organization behavior: a background - Selecting the right person for the job - Instruction in the best methods - Motivation - Stress -Health and safety -Some ethical and professional concern

UNIT V

SUGGESTED READINGS

WEB SITES
Scope: This course covers the fundamental components of the Ruby Programming Language. Emphasis is placed on the object-oriented aspects of Ruby. Topics include arrays, hashes, regular expressions, I/O, exceptions, modules, and applications areas. Ruby is a programming language with a focus on simplicity and productivity.

Objective: To help students to
- Develop server-side Ruby scripts for publishing on the Web
- Employ control structures, methods, procs, arrays and hashes to create Ruby programs
- Distinguish and use various Ruby datatypes
- Master the use of arrays and hashes
- Use the extensive pre-bundled classes
- Use the I/O facilities of Ruby to read and write binary and text files
- Master the use of Iterators to loop through various data structures
- Use Exceptions in handling various run-time errors
- Create Ruby modules
- Use the wide variety of Ruby Modules that come with the Ruby distribution
- Use object-oriented programming conventions to develop dynamic interactive Ruby applications

UNIT I

UNIT II

UNIT III
UNIT IV

UNIT V

SUGGESTED READINGS:

WEB SITES:
2. www.fincher.org/tips/Languages/Ruby
3. www.troubleshooters.com/codecom/ruby/basictutorial.htm
Instruction Hours / week: L: 4 T: 0 P: 0  Marks: Internal: 40 External: 60 Total: 100  End Semester Exam: 3 Hours

Scope: This course will discuss a variety of frameworks for identifying information technology applications. The scope of IT applications would cover Management Information System, Decision Support System, Executive Information System, and Expert System.

Objective: Provide students with a broad understanding of the types of the benefits information technology applications can provide in an organization through transaction processing, management and operational control, decision support systems, office automation, organizational communications and group work support.

UNIT I

UNIT II
Benefits of information technology applications in an organization through transaction processing, management and operational control, decision support systems, office automation, organizational communications and group work support.

UNIT III
Socio-economic environment and information systems in organization and the impact of information systems on organizations markets; frameworks for information systems planning, information systems, and competitive advantage; the new strategic role of information systems: methodologies for evaluating investments in IT; frameworks and methodologies - case studies.

UNIT IV
Design of reporting system including a discussion of principles in indicator design; managing information support activity in organizations; concept of the business process re-engineering (BPR) and how IT can enable BPR.

UNIT V
Critical success factor in implementing IT applications including the need for managing the process of change illustrated through case studies of successful/failed IT projects. Critical role of security in implementing IT applications should be discussed.
SUGGESTED READINGS

Scope: This scope of this course to explain the students the fundamentals of big data analytics and the methodologies used in storing, manipulating, and analyzing big data.

Objectives: To impart to students the skills required to design scalable systems that can accept, store, and analyze large volumes of unstructured data.

UNIT-I

UNIT-II

UNIT-III
Hadoop - Hadoop Distributed File System - Hadoop MapReduce- The Hadoop foundation and Ecosystem.

UNIT-IV
Big Data Analytics-Text Analytics and Big Data-Customized Approaches for Analysis of Big Data

UNIT-V
Integrating Data Sources-Real-Time Data Streams and Complex Event Processing-Operationalizing Big Data.

SUGGESTED READINGS


WEBSITES

1. www.oracle.com/BigData
4. www.solacesystems.com
5. en.wikipedia.org/wiki/Big_data
**Scope:** This course helps to distinguish between local programming (on a single machine) and distributed programming using multiple components via a network. This course tends to cover what designers and programmers need to consider in developing applications in a distributed parallel computing environment.

**Objective:** To help students to
- Understand the architecture and topology of network
- Understand the design process of a distributed systems
- Examine distributed and parallel computing operating system
- Know the need and challenges of distributed database

**UNIT – I**

**UNIT – II**
Message ordering and Group Communication: message ordering paradigm – Asynchronous execution with synchronous communication – classification of application level multicast algorithm – distributed multicast algorithm at the network layer.

**UNIT – III**

**UNIT – IV**

**UNIT – V**
**SUGGESTED READINGS**


**WEB SITES**

- wikipedia.org/wiki/Distributed_computing
- www.webopedia.com/TERM/D/distributed_computing.html
- www.tech-faq.com/distributed-computing.shtml
Scope:
Software metrics provide a quantitative basis for the development and validation of models of the software development process. Metrics can be used to improve software productivity and quality.

Objective: To help students to
- Understand the basic terminology and state fundamental facts about software metrics and process models.
- Identify the essential elements of a given metric or model, describe the interrelationships among its various elements
- Understand software process assessment cycles, complexity metrics and models.

UNIT I

UNIT-II
Applying the seven basic quality tools in software development-Defact removal effectiveness-The rayleigh model-Exponential distribution and reliability growth models-Quality management models

UNIT-III
In-process metrics for software testing-Complexity metrics and models-Metrics and lessons learned for object oriented projects-Availability metrics

UNIT-IV
Measuring and analyzing customer satisfaction-Conducting in-process quality assessments

UNIT-V
Software project assessments-Dos and don’t of software process improvement-Using function point metrics to measure software process improvement-Concluding remarks

SUGGESTED READINGS:
3. C. Ravindranath Pandian.(2003), Software Metrics: A Guide to Planning, Analysis, and Application,
Scope: This course provides the methods to discover, classify and build ontology for searching. To build and implement a small ontology that is semantically descriptive of chosen problem domain. To implement applications that can access, use and manipulate the ontology.

Objectives:
- To represent data from a chosen problem in XML with appropriate semantic tags obtained or derived from the ontology.
- To understand the semantic relationships among these data elements using Resource.
- To design and implement a web services application that “discovers” the data and/or other Description Framework (RDF).web services via the semantic web. Able to discover the capabilities and limitations of semantic web technology for many applications.

UNIT I

UNIT II

UNIT III

UNIT IV
UNIT V
Semantic Web Tools And Applications  

SUGGESTED READINGS
Scope: This course will lay emphasis on the practical aspects with a solid knowledge base. Typical real life situations are covered under the syllabus. The course covers topics of how to compute taxable income under the heads - Salaries, House properties, Business and Profession, Capital gains and other sources.

Objectives: The objective of the course is to give the students an idea about Indirect Taxes - Excise duty, VAT, Service tax etc. and tax planning which attempts to reduce tax liability legally. This course is intended to give a spurt to aspiring tax enthusiasts who have a natural flair to develop tax base. After completion of this course, students can start his/her career as a tax consultant or tax practitioner; thereby creating opportunities for self-employment.

UNIT I

UNIT II

UNIT III

UNIT IV
Wealth Tax: Chargeability, valuation, return, appeals, revisions, payment and recovery, gift tax: chargeability, rebate, assessment, appeals, revisions, payment and recovery.

UNIT V

SUGGESTED READINGS

1. Central and State tax acts.
1. Design an online loan application form.
2. Design a form to submit your resume in net.
3. Design an application for Library Management System
4. Design form for online reservation in air ways.
5. Design form for online shopping
6. Design an application for creating an online Advertisement.
7. Design an application for student Information System.
8. Design an application to display cookies information.
9. Write a program for display environment variables in MySQL.
10. Write a program to count web page hits.
11. Design an email form that validates the inputs, produces errors when inputs are typed incorrectly, and send an email to you when submitted.
12. Design an application to upload multiple files
VB.Net
1. Write a Program to perform various string manipulation functions.
2. Using windows application form, Develop a program to create a form, place controls and manipulate data.
3. Write a program to create inventory control using class library.
4. Write a program to create Web Services Using Vb.Net.
5. Write a program to create a screen saver using controls
6. Create an ActiveX program with simple example.
7. Using windows Application: Design Employee Details, use Sql Server as back end and also use checked list box.

ASP.Net
1. Write a program to create an on-line quiz using content page holder.
2. Write a program to retrieve Cookies information.
3. Write a program to count web page hits.
4. Write program to retrieve environment variables browser capability information.
5. Write a program for database connectivity to retrieve student information
1. Use the following learning schemes, with the default settings to analyze the weather data (in weather.arff). For test options, first choose "Use training set", then choose "Percentage Split" using default 66% percentage split. Report model percent error rate

2. Using iris dataset preprocess and classify it with J4.8 and Naïve Bayes Classifier. examine the tree in the Classifier output panel

3. Using the datasets ReutersCorn-Train and ReutersGrain-Train. Classify articles using binary attributes and word count attributes.

4. Apply any two association rule based algorithm for the supermarket analysis

5. Using weka Experimenter perform comparison analysis of J48, oneR and ID3 for vote dataset

6. Using Weka Experimenter perform comparison analysis of Naïve Bayes with different datasets

7. Apply ZeroR, OneR, and J48, to classify the Iris data in an experiment using 10 train and test runs, with 66% of the data used for training and 34% used for testing.

8. Using Weka Knowledge flow Sett up a flow to load an ARFF file (batch mode) and perform a cross-validation using J48 (WEKA’s C4.5 implementation).

9. Draw multiple ROC curves in the same plot window, using J48 and RandomForest as classifiers.

10. Use any three clustering algorithm on Vehicle data set and find best among them

11. Perform Preprocessing, feature selection and apply any one of the algorithm each from clustering, Association and classification to find their performance

12. Examine the performance of different filters for the breast cancer dataset
1. Simple router configuration.
2. Access and utilize the router to set basic parameters.
3. Connect, configure, and verify operation status of a device interface.
4. Implement static and dynamic addressing services for hosts in a LAN environment.
5. Identify and correct common problems associated with IP addressing and host configurations.
6. Describe basic routing concepts (including: packet forwarding, router lookup process).
7. Configure, verify, and troubleshoot RIPv2.
8. Perform and verify routing configuration tasks for a static or default route given.
9. Configure, verify and troubleshoot DHCP and DNS operation on a router.
10. Configure and verify a PPP connection between routers.
Prepare a more detailed, organized and easy-to-read documentation, for any application software, which should describe the following using Moodle tool:

1. User Requirement Documentation (USD)
2. Requirement Analysis Documentation. (RAD)
3. User Interfaces Specification. (UIS)
4. Object Oriented Design (OOD) or Low Level Design (LLD)
5. Code Documentation (CD)
6. Testing Documentation (TD)
7. User’s Guide (UG)
1. Write a ruby program to perform basic array and hash operations
2. Write a code to choose random numbers and find the behaviour of the number
3. Develop a program which gets raised when you handle an exception
4. Write a ruby code to display grade sheet of students using case.
5. Write a ruby program to evaluate polynomial.
6. Write a ruby program to draw box and fill with special characters.
7. Write a program to copy each line from input file to output file.
8. Write a program to create a button and fill the button with colors.
9. Write a program to create different color balls and make it bounce on window.
10. Write ruby program to display notebook widget.
11. Develop a ruby program to manipulate text with font color and images.
12. Write ruby program to create a main thread and execute multiple process through the main thread.
13. Write a ruby code to display color pallet and open dialog with the help of tk controls.
14. Design an application form using tk classes and validate all fields on Rails framework.
1. Create a set of slides with sound and animation using power point and to select various styles of slides from slide template.

2. Prepare an organization chart for a company and a college using power point.

3. To perform comparative study of your UG degree subject marks and create a various styles of graph using excel.

4. To perform student and hostel fee particulars in a worksheet. Analysis the results using excel.

5. To create a word document for salary preparation and access the calculation, graph from excel using OLE.

6. Apply the mail merge concept using word.

7. Design organizational hierarchy chart

8. Design power point Presentation to market a product

9. Create Tables for Inventory Control

10. Design Animation power point Presentation with audio and video to advertize a product