JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

MASTER OF COMPUTER APPLICATIONS (MCA) R17
Effective from Academic Year 2017- 18 admitted batch

COURSE STRUCTURE AND SYLLABUS

I Year – I Semester

Category	Course Title	Int. marks	Ext. marks	L	P	C
Core Course - I	Mathematical Foundations of Computer Science	25	75	4	--	4
Core Course - II	Computer Organization	25	75	4	--	4
Core Course - III	Computer Programming Using C	25	75	4	--	4
Core Course - IV	Probability and Statistics	25	75	4	--	4
Core Course - V	Accounting and Financial Management	25	75	4	--	4
Laboratory - I	IT Workshop	25	75	--	3	2
Laboratory - II	Computer Programming Using C Lab	25	75	--	3	2
Laboratory - III	Computer Organization Lab	25	75	--	3	2
Total Credits		**200**	**600**	**20**	**9**	**26**

I Year – II Semester

Category	Course Title	Int. marks	Ext. marks	L	P	C
Core Course - I	Data Structures and Algorithms	25	75	4	--	4
Core Course - II	Object Oriented programming using C++	25	75	4	--	4
Core Course - III	Scripting Languages	25	75	4	--	4
Core Course - IV	Operations Research	25	75	4	--	4
Core Course - V	Software Engineering	25	75	4	--	4
Laboratory - I	Data Structures Lab Using C++	25	75	--	3	2
Laboratory - II	Scripting(Python) Lab	25	75	--	3	2
Laboratory - III	English Language Communication Skills Lab	25	75	--	3	2
Total Credits		**200**	**600**	**20**	**9**	**26**

II Year - I Semester

Category	Course Title	Int. marks	Ext. marks	L	P	C
Core Course - I	Data Base Management Systems	25	75	4	--	4
Core Course - II	Computer Networks	25	75	4	--	4
Core Course - III	Operating Systems	25	75	4	--	4
Core Course - IV	Object Oriented Analysis and Design using UML	25	75	4	--	4
Core Course - V	Java Programming	25	75	4	--	4
Laboratory - I	Data Base Management Systems Lab	25	75	--	3	2
Laboratory - II	Java Programming Lab	25	75	--	3	2
Total Credits		**175**	**525**	**20**	**6**	**24**

II Year - II Semester

Category	Course Title	Int. marks	Ext. marks	L	P	C
Core Course - I	Web Technologies	25	75	4	--	4
Core Course - II	Linux Programming	25	75	4	--	4
Core Elective – I	1. Machine Learning 2. Data Warehousing and Data Mining 3. Information Security	25	75	4	--	4
Core Elective – II	1. Distributed Systems 2. Software Process and Project Management 3. Soft Computing	25	75	4	--	4
Open Elective – I	Open Elective	25	75	4	--	4
Laboratory - I	Web Technologies Lab	25	75	--	3	2
Laboratory - II	Linux Programming Lab	25	75	--	3	2
Total Credits		**175**	**525**	**20**	**6**	**24**

III Year - I Semester

Category	Course Title	Int. marks	Ext. marks	L	P	C
Core Course - I	Mobile Application Development	25	75	4	--	4
Core Course - II	Software Testing Methodologies	25	75	4	--	4
Core Elective – III	1. Web Services and SOA 2. Distributed Databases 3. Information Retrieval Systems	25	75	4	--	4
Core Elective – IV	1. Ethical Hacking 2. Multi Media and Rich Internet Applications 3. Semantic web and Social Networks	25	75	4	--	4
Open Elective – II	Open Elective	25	75	4	--	4
Laboratory - I	Mobile Application Development Lab	25	75	--	3	2
Laboratory - II	Software Engineering Methodologies and UML Lab	25	75	--	3	2
Total Credits		**175**	**525**	**20**	**6**	**24**

III Year - II Semester

Course Title	Int. marks	Ext. marks	L	P	C
Comprehensive Viva Voce	--	100	--		4
Project Work Review II	100	--	--	8	4
Project Viva Voce	--	100	--	20	16
Total Credits	**100**	**200**	**--**	**28**	**24**

*For Project Review I, Please refer 7.8 in R17 Academic Regulations.

❖ Basic Computer Programming skills is required for all open electives. Additionally knowledge on the specified area mentioned in prerequisites is required for opting the open elective
❖ Note: A student can register for any open elective subject provided that he has not already registered for the same subject.

S. No.	Open Electives	Prerequisites
1.	'R' Programming	Maths, Statistics
2.	Algorithmics	--
3.	Big Data Analytics	Data Bases , Maths
4.	Bioinformatics	Data Structures
5.	Biometrics	--
6.	Computer Forensics	Maths, Data Structures
7.	Cyber Security	Internet Technologies
8.	Distributed Systems Security	Information Security
9.	E-Commerce	Internet Technologies
10.	Embedded Systems	Digital logic
11.	Intellectual Property Rights	--
12.	Internet of Things	Java
13.	Internet Technologies and Services	Java
14.	Mobile Computing	Java
15.	Mobile Application Security	Mobile Application Development
16.	Open Stack Cloud Computing	Linux Programming
17.	Cryptography and Information Security	Maths
18.	Social Media Intelligence	--
19.	Storage Area Networks	Computer Networks
20.	Web Usability	--
21.	Organizational Structure and Personal Management	

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

MCA - I Year - I SEM

MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE

Course Objectives:
To develop problem-solving techniques and explore topics in a variety of areas of discrete mathematics, including but not limited to logic, graph theory, set theory, recursions, combinatorics, and algorithms. Students will learn to express statements in the language of formal logic and draw conclusions, model situations in terms of graph and set theory, find and interpret recursive definitions for mathematical sequences, use combinatorial methods to approach counting problems.

Course Outcomes:
- Ability to illustrate by examples the basic terminology of functions, relations, and sets and demonstrate knowledge of their associated operations.
- Ability to demonstrate in practical applications the use of basic counting principles of permutations, combinations, inclusion/exclusion principle and the pigeonhole methodology.
- Ability to represent and Apply Graph theory in solving computer science problems.

UNIT - I
Mathematical Logic: Statements and notations, Connectives, Well formed formulas, Truth Tables, tautology, equivalence implication, Normal forms. Theory of inference for the statement calculus: Rules of inference, Consistency, proof of contradiction, Automatic Theorem Proving. Predicate calculus: Predicative logic, Free and Bound variables, The Universe of Discourse. Inference theory of predicate calculus involving quantifiers.

UNIT - II
Relations: Properties of binary Relations, equivalence, transitive closure, compatibility and partial ordering relations, Lattices, Hasse diagram. Functions: Composition of functions, Inverse Function, Hashing functions, Natural numbers, recursive functions.
Algebraic structures: Algebraic systems, Examples and general properties, Semi groups and monoids, groups and sub groups, homomorphism. Lattice as partially ordered sets, Boolean algebra.

UNIT - III
Elementary Combinatorics: Basics of counting, Combinations & Permutations, with repetitions, Constrained repetitions, Binomial Coefficients, Binomial and Multinomial theorems, the principles of Inclusion – Exclusion, Pigeon hole principles and its application.

UNIT - IV
Recurrence Relations: Generating Functions, Function of Sequences, Calculating Coefficients of generating functions, Recurrence relations, Solving recurrence relation by substitution and Generating functions, the method of Characteristic roots, solution of Inhomogeneous Recurrence Relations.

UNIT - V
Graphs: Basic Concepts, Isomorphism and Sub graphs, Trees and their properties, Spanning Trees, Directed trees, Binary trees, Planar Graphs, Multi graphs and Euler circuits, Hamiltonian graphs, Chromatic Numbers.

TEXT BOOKS:
1. Discrete Mathematics with Applications to Computer Science, J P Trembley and R Manohar, TMH, 2008. (Units I and II)
2. Discrete Mathematics for Computer Scientists and Mathematicians, second edition, J.L. Mott, A. Kandel, T.P. Baker, PHI

REFERENCE BOOKS:
1. Elements of Discrete Mathematics- A Computer Oriented Approach, C.L.Liu, D.P. Mohapatra, 3rd edition, TMH.
2. Discrete and Combinatorial Mathematics- An Applied Introduction-5th Edition–Ralph. P. Grimaldi, Pearson Education.
3. Discrete Mathematics with applications, Thomas Koshy, Elsevier.
4. Discrete Mathematical Structures, Mallik and Sen, Cengage Learning.
5. Discrete Mathematical Structures, Bernand Kolman, Robert C. Busby, Sharon Cutler Ross, PHI.
6. Discrete Mathematics and its Applications, 6th edition, K.H.Rosen, TMH.
7. Discrete Mathematics, Lovasz, Springer.
8. Logic and Discrete Mathematics, Grass Man and Tremblay, Pearson Education.
9. Discrete Mathematics, S K Chakraborthy and B K Sarkar, Oxford, 2011.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

MCA - I Year - I SEM

COMPUTER ORGANIZATION

Course Objectives:

- To understand how computers are constructed out of a set of functional units.
- To understand how these functional units operate, interact and communicate.
- To understand the factors and trade-offs that affect computer performance.
- To understand concrete representation of data at the machine level.
- To understand how computations are actually performed at the machine level.
- To understand how problems expressed by humans are expressed as binary strings in a machine.
- Understand the system interconnection and the different I/O techniques.
- Explain the functioning and programming of the INTEL-8086.
- Understand the design of processors, the structure, and operation of memory and virtual memory, cache, storage, and pipelining, system integration, and peripherals.
- Identify the different architectural and organizational design issues that can affect the performance of a computer such as Instruction Sets design, Pipelining, RISC architecture, and Superscalar architecture.
- Design an interconnection networks and multiprocessors.

Course Outcomes: After this course students understand in a better way the I/O and memory organization in depth. They should be in a position to write assembly language programs for various applications.

UNIT - I
Number Systems and Computer Arithmetic: Signed and unsigned numbers, Addition and subtraction, multiplication, division, Floating point representation, logical operation, Gray code, BCD codes, Error detecting codes, Boolean algebra, Simplification of Boolean expressions, K-Maps, Combinational and Sequential Circuits- decoders, Encoders, Multiplexers, Half and Full adders, Shift registers, Sequential circuits- flip-flops.

UNIT - II
Memory Organization: Memory hierarchy, Main memory-RAM, ROM chips, Memory address map, memory contention to CPU, Associative Memory-Hardware logic, match, read and write logic, Cache Memory-Associative mapping, Direct mapping, Set-associative mapping, hit and miss ratio.

UNIT - III
Basic CPU Organization: Introduction to CPU, Instruction formats-INTEL-8086 CPU architecture-Addressing modes - generation of physical address- code segment registers, Zero, one, two, and three address instructions. INTEL 8086 Assembly Language Instructions-Data transfer instructions-input- output instructions, address transfer, Flag transfer, arithmetic, logical, shift, and rotate instructions. Conditional and unconditional transfer, iteration control, interrupts, and process control instructions, assembler directives, Programming with assembly language instructions.

UNIT - IV
Input -Output Organization: Peripheral devices, input-output interface-I/O Bus and interface modules, I/O versus Memory bus, isolated versus memory mapped I/O, Modes of transfer-Programmed I/O,

Interrupt-initiated I/O, priority interrupts-Daisy chaining, parallel priority, interrupt cycle, DMA- DMA control, DMA transfer, Input output processor-CPU-IOP communication.

UNIT - V

Pipeline and Vector Processing**:** Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processors.

Multi Processors**:** Characteristics or Multiprocessors, Interconnection Structures, Inter processor Arbitration, Inter Processor Communication and Synchronization, Cache Coherence, Shared Memory Multiprocessors.

TEXT BOOKS:
1. Computer System Architecture, M. Morris Mano, 3rd Edition, Pearson Education, 2008.
2. Microprocessors and Interfacing, Douglas Hall, Tata McGraw-Hill.

REFERENCE BOOKS:
1. Computer Organization, Carl Hamacher, Zvonko Vranesic, Safwat Zaky, 5th Edition, McGraw Hill.
2. Fundamentals of Computer Organization and Design, Sivarama P. Dandamudi, Springer Int. Edition.
3. Computer Organization and Architecture, William Stallings, 8th Edition, Pearson, 2007.
4. Digital Design, M. Morris Mano, Pearson Education.
5. Computer Organization and Design, D. A. Paterson and John L. Hennessy, Elsevier.
6. Computer Architecture and Organization, M. Murdocca and V. Heuring, Wiley India.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MCA - I Year - I SEM

COMPUTER PROGRAMMING USING C

Course Objectives:
- To understand the various steps in Program development.
- To understand the basic concepts in C Programming Language.
- To learn how to write modular and readable C Programs
- To learn to write programs (using structured programming approach) in C to solve problems.
- To introduce the students to basic data structures such as lists, stacks and queues.
- To make the student understand simple sorting and searching methods.

Course Outcomes:
- Demonstrate the basic knowledge of computer hardware and software.
- Ability to apply solving and logical skills to programming in C language and also in other languages.

UNIT - I
Introduction to Computers: Computer Systems, Computing Environments, Computer Languages, Creating and running programs, Program Development.
Introduction to the C Language: Background, C Programs, Identifiers, Types, Variables, Constants, Input / Output, Operators (Arithmetic, relational, logical, bitwise etc.), Expressions, Precedence and Associativity, Expression Evaluation, Type conversions.

UNIT - II
Statements- Selection Statements(making decisions) – if and switch statements, Repetition statements (loops)-while, for, do-while statements, Loop examples, other statements related to looping – break, continue, goto, Simple C Program examples.
Functions: Designing Structured Programs, Functions, user defined functions, inter function communication, Standard functions, Scope, Storage classes-auto, register, static, extern, scope rules, type qualifiers, recursion- recursive functions, Limitations of recursion, example C programs, Preprocessor commands.

UNIT - III
Arrays: Concepts, using arrays in C, inter function communication, array applications, two – dimensional arrays, multidimensional arrays, C program examples.
Pointers: Introduction (Basic Concepts), Pointers for inter function communication, pointers to pointers, compatibility, Pointer Applications-Arrays and Pointers, Pointer Arithmetic and arrays, Passing an array to a function, memory allocation functions, array of pointers, programming applications, pointers to void, pointers to functions.

UNIT - IV
Strings – Concepts, C Strings, String Input / Output functions, arrays of strings, string manipulation functions, string / data conversion, C program examples.
Enumerated, Structure ,and Union Types– The Type Definition(typedef), Enumerated types, Structures –Declaration, initialization, accessing structures, operations on structures, Complex structures, structures and functions, Passing structures through pointers, self referential structures, unions, bit fields, C programming examples, command –line arguments.

UNIT - V

Input and Output – Concept of a file, streams, text files and binary files, Differences between text and binary files, State of a file, Opening and Closing files, file input / output functions (standard library input / output functions for files), file status functions (error handling)Positioning functions, C program examples.

TEXT BOOKS:

1. A Structured Programming Approach Using C, B.A. Forouzan and R.F. Gilberg, Third Edition, Cengage Learning.
2. Programming in C. P. Dey and M Ghosh, Oxford University Press.

REFERENCE BOOKS:

1. C & Data structures – P. Padmanabham, Third Edition, B. S. Publications.
2. Problem Solving and Program Design in C.J.R. Hanly and E.B. Koffman, 7th Edition, Pearson education.
3. Programming in C. Ajay Mittal, Pearson.
4. Programming with C. B. Gottfried, 3rd edition, Schaum's outlines, TMH.
5. Problem solving with C, M.T. Somasekhara, PHI.
6. Programming with C, R.S. Bickar, Universities Press.
7. Computer Programming & Data Structures, E. Balagurusamy, 4th edition, TMH.
8. Programming in C – Stephen G. Kochan, III Edition, Pearson Education.
9. The C Programming Language, B.W. Kernighan and Dennis M.Ritchie, PHI.
10. C Programming with problem solving, J.A. Jones & K. Harrow, Dreamtech Press.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MCA - I Year - I SEM

PROBABILITY AND STATISTICS

Course Objectives:
- Understand the concepts of Probability, random variables and their distributions.
- Understand the concepts of estimation and hypothesis testing for population averages and percentages.
- Select and produce the appropriate tabular and graphical formulas for displaying bivariate data sets and carry out correlation, regression and chi-square analyses.

UNIT - I
Probability: Sample space and events – Probability – The axioms of probability - Some elementary theorems - Conditional probability – Baye's theorem.

UNIT - II
Random variables – Discrete and continuous – Distribution – Distribution function.
Distribution - Binomial, Poisson and normal distribution – related properties.

UNIT - III
Sampling distribution: Populations and samples - Sampling distributions of mean (known and unknown) proportions, sums and differences.
Estimation: Point estimation – interval estimation - Bayesian estimation.

UNIT - IV
Test of Hypothesis – Means and proportions – Hypothesis concerning one and two means – Type I

and Type II errors. One tail, two-tail tests. Tests of significance – Student's t-test, F-test, χ^2 test. Estimation of proportions.

UNIT - V
Curve fitting: The method of least squares – Inferences based on the least squares estimations - Curvilinear regression – multiple regressions – correlation for univariate and bivariate distributions.

TEXT BOOKS:
1. Probability and statistics for engineers (Erwin Miller and John E. Freund), R A Johnson and C.B.Gupta.7th edition, PHI.
2. Introduction to Probability and Statistics, 12th edition, W. Mendenhall, R.J. Beaver and B.M. Beaver, Cengage Learning.

REFERENCE BOOKS:
1. Text book of Probability and Statistics Dr. Shahnaz Bathul, V.G.S. Publishers 2003.
2. Probability and Statistics in Engineering, 4th Edition, William W. Hines, Douglas C. Montgomery, David M. Goldsman, Connie M.Borror, Wiley Student Edition.
3. Probability, Statistics and Queuing Theory, 2nd Edition, Trivedi, John Wiley and Sons
4. Introduction to Probability and Statistics, J.S. Milton, Jesse C. Arnold, 4th edition, TMH.
5. Probability, Statistics and Random Processes, Dr. K. Murugesan, P. Guruswamy, Anuradha Agencies, Deepti Publications.
6. Probability and Statistics for Engineers, G.S.S. Bhisma Rao, Sitech, 2nd edition, 2005.
7. Probability and Statistics for Engineers and Scientists, R. E. Walpole, S.L. Myers, K.Ye, Pearson.

8. Probability, Statistics and Reliability for Engineers and Scientists, B. M. Ayyub, R.H. McCuen, 2nd edition, Chapman & Hall/CRC, Special Indian Edition.
9. Probability and Statistics for Engineers and Scientists, Sheldon M Ross, 4th ed., Elsevier, 2011.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MCA - I Year - I SEM

ACCOUNTING AND FINANCIAL MANAGEMENT

This course is designed to introduce students to the principles, concepts, and applications of financial accounting and management.

An introduction to accounting emphasizing how general purpose financial statements communicate information about the business corporation's performance and position for users internal and external to management. Approximately one third of the course emphasizes how the accountant processes and presents the information and includes exposure to recording transactions, adjusting balances and preparing financial statements for service and merchandise firms according to established rules and procedures.

This course presents the underlying framework and concepts of Financial Accounting in the context of how accounting fits into the overall business environment of contemporary society. Financial accounting is the basic means of recording and reporting financial information in a business. Students will learn how accounting functions as an information development and communication system that supports economic decision making and provides value to entities and society. Students will discover the uses and limitations of financial statements and related information and apply analytical tools in making both business and financial decisions. Topics examined include those related to corporate financial position, operating results, cash flows, and financial strength. Students will study the basic accounting system and will be shown how the various accounting alternatives for recording financial transactions impact on the usefulness of the information provided for decision-making. During coverage of relevant topics reference will be made to recent lapses in ethical reporting and the resulting impact on the financial markets and society.

In Financial Management students are introduced to concepts and tools that enable them to think critically about the financial opportunities and challenges faced by an organization. In working through the material, they build upon the understanding they already have of accounting, business, and mathematics. They learn how to use financial statements such as balance sheets, income statements, and statements of cash flow. They prepare budgets, analyze investment options, and determine the best means of financing business endeavors. They discover ways of assessing both the return and the risk involved in a firm's financial decisions. The focus of this course is on solving practical business problems similar to those encountered in the workplace.

Course Objectives

- Explain financial accounting terminology and the recording process.
- Discuss and apply the basic principles of accounting, the accounting model, and the accounting cycle.
- Demonstrate and show how items such as assets, liabilities, share capital, etc., are reported in the financial statements.
- Carry out journal entries and adjustments.
- Prepare and analyze a trial balance and financial statements.
- Analyze a cash flow statement.
- Analyze financial statements using ratio analysis.
- Understand the elements, uses, and limitations of each financial statement and the relationships among the statements
- Understand how financial information, primarily that provided by the financial statements, can be used to analyze business operations and make economic decisions
- Identify the basic principles used in safeguarding assets and insuring the accuracy of accounting records

- Appreciate the role accounting software applications play in gathering, recording, reporting and interpreting financial accounting information
- Interpret and create standard financial statements.
- Perform financial statement analysis for the purposes of evaluating and forecasting in financial management.
- Evaluate a firm's working capital position.
- Estimate the components of cost of capital by applying time value of money principles.
- Perform net present value analysis for capital budgeting purposes.
- Evaluate risk in the capital budgeting process.
- Evaluate the cost of debt, preferred stock, and common stock as sources of capital.
- The Most important learning objective is how to do accounts in Computers.

Prerequisites:
Some basic accounting knowledge and mathematics level is recommended.
Ability to understand the knowledge of cost estimation towards the business sceneries.
Ability to analyze market competency in the global village
To become a Master in accounting and financial management.

UNIT - I
Introduction to Accounting: Principles, concepts and conventions, double entry system of accounting, introduction to basic books of accounts, Journal, ledger- Trial Balance - Preparation of Final accounts: Trading Account, Profit and Loss Account and Balance Sheet.

UNIT - II
Financial Management - Meaning and scope, role of Financial Manager, Objectives of time value of money - Goals of Financial Management, Leverages: Operating, Financial Leverage and Combined Leverage Cost of Capital: Cost of Equity, Preference Shares, Bonds- Weighted Average Cost of Capital – Capital Gearing- Overcapitalization and Undercapitalization, Sources of Finance.

UNIT - III
Tools and Techniques for Financial Statement Analysis: Ratio Analysis – Classification of Ratios –Short term solvency and long term solvency – Profitability ratios - Analysis and Interpretation of Financial Statements through ratios of Liquidity, Solvency and Profitability ratios. **Fund Flow Statement** - Meaning, Importance, Statement of changes in working capital and statement of Sources and application of funds. Cash flow Analysis: cash flow Statements: Preparation, Analysis and interpretation.

UNIT - IV
Break-even Analysis: Concept of Break Even Point, Cost-Volume-Profit Analysis, Determination of Break Even Point, Margin of Safety and PV ratio, Impact of changes in Cost or selling price on BEP Practical applications of Break-even Analysis.
Budgeting: Budgeting–cash budget, sales budget – flexible Budgets and master budgets.

UNIT - V
Capital Budgeting: Capital and its significance, Types of Capital, Estimation of Fixed and Working capital requirements, Methods and sources of raising capital.. Capital Budgeting: features of capital budgeting proposals, Methods of Capital Budgeting: Payback Method, Accounting Rate of Return (ARR) and Net Present Value Method (simple problems).

TEXT BOOKS:
1. Aryasri: Accounting And Financial Management,, TMH, 2009
2. Van Horne, James, C: Financial Management and Policy, Pearson, 2009

REFERENCE BOOKS:
1. Dr. G. Vidyanath G. Lakshmi Accounting and Financial Management.
2. Prasanna Chandra, Financial Management, TMH, 2009
3. S. N. Maheshwari, Financial Accounting, Sultan Chand, 2009.
4. Tulsian, Financial Accounting, S Chand, 2009.
5. Khan and Jain: Financial Management, TMH, 2009
6. Gokul Sinha: Financial Statement Analysis, PHI, 2009
7. Bhat Sundhindra: Financial Management, Excel:2009
8. Jawaharlal: Accounting for Management, Himalaya, 2009
9. Paresh Shah : Basic Financial Accounting for Management, Oxford 2010.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MCA - I Year - I SEM

IT WORKSHOP

Course Objectives:
The IT Workshop for engineers is a training lab course spread over 54 hours. The modules include training on PC Hardware, Internet & World Wide Web and Productivity tools including Word, Excel and Power Point.

PC Hardware introduces the students to a personal computer and its basic peripherals, the process of assembling a personal computer, installation of system software like MS Windows, Linux and the required device drivers. In addition hardware and software level troubleshooting process, tips and tricks would be covered. **The students should work on working PC to disassemble and assemble to working condition and install Windows and Linux on the same PC. Students are suggested to work similar tasks in the Laptop scenario wherever possible.**

Internet & World Wide Web module introduces the different ways of hooking the PC on to the Internet from home and workplace and effective usage of the internet. Usage of web browsers, email, newsgroups and discussion forums would be covered. In addition, awareness of cyber hygiene, i.e., protecting the personal computer from getting infected with the viruses, worms and other cyber attacks would be introduced.

Productivity tools module would enable the students in crafting professional word documents, excel spread sheets and power point presentations using the Microsoft suite of office tools and LaTeX. (**Recommended to   use Microsoft office 2007 in place of MS Office 2003**)

PC Hardware
Task 1 : Identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram of the CPU along with the configuration of each peripheral and submit to your instructor.

Task 2 : Every student should disassemble and assemble the PC back to working condition. Lab instructors should verify the work and follow it up with a Viva. Also students need to go through the video which shows the process of assembling a PC. A video would be given as part of the course content.

Task 3 : Every student should individually install MS windows on the personal computer. Lab instructor should verify the installation and follow it up with a Viva.

Task 4 : Every student should install Linux on the computer. This computer should have windows installed. The system should be configured as dual boot with both windows and Linux. Lab instructors should verify the installation and follow it up with a Viva.

Task 5: Hardware Troubleshooting: Students have to be given a PC which does not boot due to improper assembly or defective peripherals. They should identify the problem and fix it to get the computer back to working condition. The work done should be verified by the instructor and followed up with a Viva.

Task 6: Software Troubleshooting: Students have to be given a malfunctioning CPU due to system software problems. They should identify the problem and fix it to get the computer back to working condition. The work done should be verified by the instructor and followed up with a Viva.

Internet & World Wide Web

Task 1: Orientation & Connectivity Boot Camp: Students should get connected to their Local Area Network and access the Internet. In the process they configure the TCP/IP setting. Finally students should demonstrate, to the instructor, how to access the websites and email. If there is no Internet connectivity preparations need to be made by the instructors to simulate the WWW on the LAN.

Task 2: Web Browsers, Surfing the Web: Students customize their web browsers with the LAN proxy settings, bookmarks, search toolbars and pop up blockers. Also, plug-ins like Macromedia Flash and JRE for applets should be configured.

Task 3: Search Engines & Netiquette: Students should know what search engines are and how to use the search engines. A few topics would be given to the students for which they need to search on Google. This should be demonstrated to the instructors by the student.

Task 4: Cyber Hygiene: Students would be exposed to the various threats on the Internet and would be asked to configure their computer to be safe on the Internet. They need to first install antivirus software, configure their personal firewall and windows update on their computer. Then they need to customize their browsers to block pop ups, block active x downloads to avoid viruses and/or worms.

Task 5: Develop your home page using HTML Consisting of your photo, name, address and education details as a table and your skill set as a list.

Productivity tools:

LaTeX and Word:
Word Orientation: The mentor needs to give an overview of LaTeX and Microsoft (MS) office 2007/ equivalent (FOSS) tool word: Importance of LaTeX and MS office 2007/ equivalent (FOSS) tool Word as word Processors, Details of the three tasks and features that would be covered in each, using LaTeX and word – Accessing, overview of toolbars, saving files, Using help and resources, rulers, format painter.

Task 1: Using LaTeX and Word to create project certificate. Features to be covered:-Formatting Fonts in word, Drop Cap in word, Applying Text effects, Using Character Spacing, Borders and Colors, Inserting Header and Footer, Using Date and Time option in both LaTeX and Word.

Task 2 : Creating project abstract Features to be covered:-Formatting Styles, Inserting table, Bullets and Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check, Track Changes.

Task 3 : Creating a Newsletter : Features to be covered:- Table of Content, Newspaper columns, Images from files and clipart, Drawing toolbar and Word Art, Formatting Images, Textboxes, Paragraphs and Mail Merge in word.

Excel:
Excel Orientation: The mentor needs to tell the importance of MS office 2007/ equivalent (FOSS) tool Excel as a Spreadsheet tool, give the details of the two tasks and features that would be covered in each. Using Excel – Accessing, overview of toolbars, saving excel files, Using help and resources.

Task 1: Creating a Scheduler - Features to be covered: - Gridlines, Format Cells, Summation, auto fill, Formatting Text.

Task 2 : Calculating GPA - .Features to be covered:- Cell Referencing, Formulae in excel – average, std. deviation, Charts, Renaming and Inserting worksheets, Hyper linking, Count function, LOOKUP/VLOOKUP, Sorting, Conditional formatting.

LaTeX and MS/equivalent (FOSS) tool Power Point:
Task1: Students will be working on basic power point utilities and tools which help them create basic power point presentation. Topic covered during this week includes :- PPT Orientation, Slide Layouts, Inserting Text, Word Art, Formatting Text, Bullets and Numbering, Auto Shapes, Lines and Arrows in both LaTeX and Power point. Students will be given model power point presentation which needs to be replicated (exactly how it's asked).

Task 2: Second week helps students in making their presentations interactive. Topic covered during this week includes: Hyperlinks, Inserting –Images, Clip Art, Audio, Video, Objects, Tables and Charts.

Task 3: Concentrating on the in and out of Microsoft power point and presentations in LaTeX. Helps them learn best practices in designing and preparing power point presentation. Topic covered during this week includes: - Master Layouts (slide, template, and notes), Types of views (basic, presentation, slide slotter, notes etc), Inserting – Background, textures, Design Templates, Hidden slides.

REFERENCE BOOKS:
1. Introduction to Information Technology, ITL Education Solutions limited, Pearson Education.
2. LaTeX Companion – Leslie Lamport, PHI/Pearson.
3. Introduction to Computers, Peter Norton, 6/e Mc Graw Hill.
4. Upgrading and Repairing, PC's 18th e, Scott Muller QUE, Pearson Education.
5. Comdex Information Technology course tool kit Vikas Gupta, WILEY Dreamtech.
6. IT Essentials PC Hardware and Software Companion Guide Third Edition, David Anfinson and Ken Quamme. – CISCO Press, Pearson Education.
7. PC Hardware and A+ Handbook – Kate J. Chase PHI (Microsoft).

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MCA - I Year - I SEM

COMPUTER PROGRAMMING USING C LAB

Course Objectives:
- To write programs in C to solve the problems.
- To implement linear data structures such as lists, stacks, queues.
- To implement simple searching and sorting methods.

Course Outcomes:
- Ability to write programs for different kinds of problems in C.
- Work confidently in compilers like C and others.
- Ability to identify appropriate data type or data structure to given problem.

Recommended Systems/Software Requirements:
- Intel based desktop PC.
- ANSI C Compiler with Supporting Editors.

Exercise-I
a) Write a C program to find the sum of individual digits of a positive integer.
b) A Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C program to generate the first n terms of the sequence.
c) Write a C program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.

Exercise-2
a) Write a C program to calculate the following Sum:
Sum=1-x^2/2! +x^4/4!-x^6/6!+x^8/8!-x^{10}/10!
b) Write a C program to find the roots of a quadratic equation.

Exercise-3
a) The total distance travelled by vehicle in 't' seconds is given by distance = ut+1/2at^2 where 'u' and 'a' are the initial velocity (m/sec.) and acceleration (m/sec^2). Write C program to find the distance travelled at regular intervals of time given the values of 'u' and 'a'. The program should provide the flexibility to the user to select his own time intervals and repeat the calculations for different values of 'u' and 'a'.
b) Write a C program, which takes two integer operands and one operator from the user, performs the operation and then prints the result. (Consider the operators +,-,*, /, % and use Switch Statement).

Exercise-4
a) Write C programs that use both recursive and non-recursive functions
        i) To find the factorial of a given integer.
        ii) To find the GCD (greatest common divisor) of two given integers.

Exercise-5
a) Write a C program to find the largest integer in a list of integers.
b) Write a C program that uses functions to perform the following:
        i) Addition of Two Matrices.
        ii) Multiplication of Two Matrices.

Exercise-6
a) Write a C program that uses functions to perform the following operations:
      i) To insert a sub-string in to a given main string from a given position.
      ii) To delete n Characters from a given position in a given string.
b) Write a C program to determine if the given string is a palindrome or not.

Exercise-7
a) Write a C program that displays the position or index in the string S where the string T begins, or –1 if S doesn't contain T.
b) Write a C program to count the lines, words and characters in a given text.

Exercise-8
a) Write a C program to generate Pascal's triangle.
b) Write a C program to construct a pyramid of numbers.

Exercise-9
Write a C program to read in two numbers, x and n, and then compute the sum of this geometric progression:
$1+x+x^2+x^3+\ldots\ldots\ldots+x^n$
For example: if n is 3 and x is 5, then the program computes 1+5+25+125.
Print x, n, the sum
Perform error checking. For example, the formula does not make sense for negative exponents – if n is less than 0. Have your program print an error message if n<0, then go back and read in the next pair of numbers of without computing the sum. Are any values of x also illegal? If so, test for them too.

Exercise-10
a) 2's complement of a number is obtained by scanning it from right to left and complementing all the bits after the first appearance of a 1. Thus 2's complement of 11100 is 00100. Write a C program to find the 2's complement of a binary number.
b) Write a C program to convert a Roman numeral to its decimal equivalent.

Exercise-11
Write a C program that uses functions to perform the following operations:
      i) Reading a complex number.
      ii) Writing a complex number.
      iii) Addition of two complex numbers.
      iv) Multiplication of two complex numbers.
(Note: represent complex number using a structure.)

Exercise-12
a) Write a C program which copies one file to another.
b) Write a C program to reverse the first n characters in a file.
(Note: The file name and n are specified on the command line.)

Exercise-13
a) Write a C program to display the contents of a file.
b) Write a C program to merge two files into a third file (i.e., the contents of the first file
    followed by those of the second are put in the third file).

Exercise-14
a) Write a C program that uses non recursive function to search for a Key value in a given list of integers using Linear search.

b) Write a C program that uses non recursive function to search for a Key value in a given sorted list of integers using Binary search.

Exercise-15
Write a C program to implement the Lagrange interpolation.

Exercise-16
Write C program to implement the Newton- Gregory forward interpolation.

Exercise-17
Write a C program to implement Trapezoidal method.

Exercise-18
Write a C program to implement Simpson method.

TEXT BOOKS:
1. C programming and Data Structures, P. Padmanabham, Third Edition, BS Publications.
2. Computer Programming in C, V. Rajaraman, PHI.
3. C Programming, E.Balagurusamy, 3rd edition, TMH.
4. C Programming, M.V.S.S.N.Prasad, ACME Learning Pvt. Ltd.
5. C and Data Structures, N.B.Venkateswarlu and E.V.Prasad, S.Chand.
6. Mastering C, K.R. Venugopal and S.R. Prasad, TMH Publications.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MCA - I Year - I SEM

COMPUTER ORGANIZATION LAB

Course Objective:
- To write assembly language programs (8086 processor) for solving problems.

List of Sample Problems:

Write assembly language programs for the following using MASAM:
1. Write assembly language programs to evaluate the expressions:
        i) $a = b + c - d * e$
        ii) $z = x * y + w - v + u / k$
     a. Considering   8-bit, 16 bit and 32 bit binary numbers as b, c, d, e.
     b. Considering 2 digit, 4digit and 8 digit BCD numbers.
       Take the   input in consecutive memory locations and results also.
       Display the results by using "int xx" of 8086. Validate program for the
       Boundary conditions.

2. Write an ALP of 8086 to add two exponential numbers which are in IEEE 754
Notation. Display the results by using "int xx" of 8086. Validate program for the boundary Conditions.

3. Write an ALP of 8086 to take N numbers as input. And do the following operations on them.
a) Arrange in ascending and descending order.
b) Find max and minimum
c) Find average
      Considering 8-bit, 16 bit binary numbers and 2 digit, 4digit and 8 digit BCD numbers.
      Display the results by using "int xx" of 8086. Validate program for the boundary conditions.

4. Write an ALP of 8086 to take a string of as input (in 'C' format) and do the Following Operations on it.
a) Find the length .   b) Find it is Palindrome or not.
c) Find whether given string substring or not.
d) Reverse a string.   e) Concatenate by taking another sting.
             Display the results by using "int xx" of 8086.

5. Write the ALP to implement the above operations as procedures and call from the main procedure.

6. Write an ALP of 8086 to find the factorial of a given number as a Procedure and call from the main program which display the result.

7. Write an assembly language program to encrypt digits as shown below:
    Input digit: 0 1 2 3 4 5 6 7 8 9
    Encrypted digit: 4 6 9 5 0 3 1 8 7 2
    Your program should accept a string consisting of digits. The encrypted
    String should be displayed using "int xx" of 8086.

8. Write a procedure to locate a character in a given string. The procedure receives a

pointer to a string and character to be located. When the first occurrence of the character is located, its position is returned to main. If no match is found, a negative value is returned. The main procedure requests a character string and a character to be located and displays the result.

9. Write an assembly language program to read a string of characters from the user and
   that prints the vowel count . Display the results by using "int xx" of 8086.

ex.   Input: Advanced Programming in UNIX
            Out put:
              Vowel   count
              a or A   3
              e or E   1
               i or I   3
                o or O   1
              u or U   1

10. A computer uses RAM chips of 1024 X 1 capacity.
a) How many chips are needed, and how should their address lines be connected to provide a memory capacity of 1024 bytes?
b) How many chips are needed to provide a memory capacity of 16K bytes?

11. A computer employs RAM chips of 256X8 and ROM chips of 1024 X 8. The computer
     needs 2K bytes of RAM, 4K bytes of ROM, and four interface units, each with four
     registers. A memory-mapped I/O configuration is used. The two highest-order bits of the
     address bus are assigned 00 for RAM, 01 for ROM, 10 for interface registers.
a. How many RAM and ROM chips are needed?
b. Draw a memory-address map for the system.
c. Give the address range in hexadecimal for RAM, ROM and interface.

12. Obtain the complement function for the match logic of one word in an associative
     memory. Draw the logic diagram for it and compare with the actual match logic diagram.

13. A two-way set associative cache memory uses blocks of four words. The cache can
     accommodate a total of 2048 words from main memory. The main memory size is 128K X 32.
a. Formulate all pertinent information required to construct the cache memory.
b. What is the size of the cache memory?

14. A digital computer has a memory unit of 64K X 16 and a cache memory of 1K words.
        The cache uses direct mapping with a block size of four words.
a. How many bits are there in each word of cache, and how are they divided
    into functions? Include a valid bit.
        b. How many bits are there in the tag, index, block, and word fields of the
   address format?
        c. How many blocks can the cache accommodate?

15. An address space is specified by 24 bits and the corresponding memory space by 16 bits.
a. How many words are there in the address space?
b. How many words are there in the memory space?
c. If a page consists of 2K words, how many pages and blocks are there in the system.

16. A virtual memory has a page size of 1K words. There are eight pages and four blocks.
     The associative memory page table contains the following entries. Make a list of all virtual addresses (in decimal) that will cause a page fault.

Page	Block
0	3
1	1
4	2
6	0

TEXT BOOKS:

1. IBM PC Assembly Language and Programming, P. Abel, 5th Edition, PHI.
2. Introduction to Assembly Language Programming, Sivarama P. Dandamudi, Springer Int. Edition, 2003.
3. The 8088 and 8086 Microprocessors: Programming, Interfacing, Software, Hardware and Application, 4th edition, W. A. Triebel, A. Singh, N. K. Srinath, Pearson Education.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS I Year II Semester

Data Structures & Algorithms

Prerequisites
- A course on "Computer Programming & Data Structures"

Course Objectives:
- Introduces the notations for analysis of the performance of algorithms
- Introduces a variety of data structures such as hash tables, disjoint sets and Priority Queue
- Describes major algorithmic techniques (divide-and-conquer, backtracking, dynamic programming and greedy) and mention problems for which each technique is appropriate;
- Introduces sorting, searching and pattern matching algorithms

Course Outcomes:
- Ability to analyze the performance of algorithms
- Ability to select the data structures that efficiently model the information in a problem
- Ability to choose appropriate data structures and algorithm design methods for a specified application
- Ability to understand how the choice of data structures and the algorithm design methods impact the performance of programs
- Design programs using a variety of data structures, including hash tables, disjoint sets, trees and graphs
- Implement and know the application of algorithms for sorting and pattern matching

UNIT - I
Introduction: Algorithm, Psuedo code for expressing algorithms, Performance Analysis-Space complexity, Time complexity, Asymptotic Notations, Substitution method, Master's theorem
Divide and conquer: General method, applications-Binary search, Quick sort, Merge sort, Strassen's matrix multiplication

UNIT - II
Disjoint Sets: Disjoint set operations, union and find algorithms
Greedy method: General method, applications-Job sequencing with deadlines, 0/1 knapsack problem, Minimum cost spanning trees, Single source shortest path problem.
Priority Queues – Definition, Realizing a Priority Queue using Heaps, operations of priority queue

UNIT – III
Dynamic Programming: General method, applications- Optimal binary search trees, 0/1 knapsack problem, All pairs shortest path problem, Travelling sales person problem, Reliability design.
Backtracking: General method, applications-n-queens problem, sum of subsets problem, graph coloring.

UNIT – IV
Review of basic data structures: The list, Stack, Queue
linear list representation, skip list representation, operations - insertion, deletion and searching.
Hash table representation: hash functions, collision resolution-separate chaining, open addressing-linear probing, quadratic probing, double hashing, rehashing, extendible hashing, comparison of hashing and skip lists.
Searching: Linear Search and Binary Search
Sortings: Bubble sort, Selection sort, Insertion sort, Radix sort, Heap sort

UNIT – V

Trees: Definition, Definition, Types of trees, Binary Trees, Binary Tree Traversal Methods, Binary search tree operations, operations of AVL tree, B-Tree, Red-Black tree, Splay tree

Graphs: Definition, Representation of graphs, Graphs Traversal Methods.

String Matching algorithms: Brute Force algorithm, Boyer Moore algorithm, Knuth- Morris-Pratt algorithm

TEXT BOOKS:
1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharam, Universities Press.
2. Fundamentals of Data Structures, Ellis Horowitz, Satraj Sahni and Rajasekharam, Universities Press.

REFERENCES:
1. Introduction to Algorithms, second edition, T. H. Cormen, C. E. Leiserson, R. L. Rivest and C.Stein,PHI Pvt. Ltd./ Pearson Education.
2. Data structures using C and C++, Langsam, Augenstein and Tanenbaum, PHI.
3. Data structures: A Pseudocode Approach with C, 2 nd edition, R. F. Gilberg and B.A. Forouzan, Cengage Learning.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS I Year II Semester

OBJECT ORIENTED PROGRAMMING THROUGH C++

UNIT - I

C++ Basics: Structure of a C++ program, Data types, Declaration of variables, Expressions, Operators, Operator Precedence, Evaluation of expressions, Type conversions, Pointers, Arrays, Pointers and Arrays, Strings, Structures, References.

Flow control statement- if, switch, while, for, do, break, continue, goto statements.

Functions - Scope of variables, Parameter passing, Default arguments, inline functions, Recursive functions, Pointers to functions.

Dynamic memory allocation and deallocation operators-new and delete, Preprocessor directives.

UNIT - II

Different paradigms for problem solving, need for OOP, differences between OOP and Procedure oriented programming, Abstraction, Overview of OOP principles, Encapsulation, Inheritance and Polymorphism.

C++ Classes And Data Abstraction: Class definition, Class structure, Class objects, Class scope, this pointer, Friends to a class, Static class members, Constant member functions, Constructors and Destructors, Dynamic creation and destruction of objects, Data abstraction, ADT and information hiding.

UNIT- III

Inheritance: Defining a class hierarchy, Different forms of inheritance, Defining the Base and Derived classes, Access to the base class members, Base and Derived class construction, Destructors, Virtual base class.

Virtual Functions And Polymorphism: Static and Dynamic bindings, Base and Derived class virtual functions, Dynamic binding through virtual functions, Virtual function call mechanism, Pure virtual functions, Abstract classes, Implications of polymorphic use of classes, Virtual destructors.

UNIT - IV

C++ I/O: I/O using C functions, Stream classes hierarchy, Stream I/O, File streams and String streams, Overloading << and >> operators, Error handling during file operations, Formatted I/O.

UNIT - V

Exception Handling: Benefits of exception handling, Throwing an exception, The try block, Catching an exception, Exception objects, Exception specifications, Stack unwinding, Rethrowing an exception, Catching all exceptions, Design issues in exception handling.

Text Books:
1. Problem solving with C++: The Object of Programming, 4th Edition, Walter Savitch, Pearson Education.
2. C++, the Complete Reference, 4th Edition, Herbert Schildt, TMH.

REFERENCES:
1. C++ Primer, 3rd Edition, S. B. Lippman and J. Lajoie, Pearson Education.
2. The C++ Programming Language, 3rd Edition, B. Stroutstrup, Pearson Education.
3. OOP in C++, 3rd Edition, T. Gaddis, J. Walters and G. Muganda, Wiley DreamTech Press.
4. Object Oriented Programming in C++, 3rd Edition, R. Lafore, Galigotia Publications pvt ltd.
5. Computer Science, a Structured Programming Approach Using C++, B. A. Forouzan and R. F. Gilberg, Thomson.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS I Year II Semester

SCRIPTING LANGUAGES

Prerequisites:
- A course on "Computer Programming and Data Structures"
- A course on "Object Oriented Programming Concepts"

Course Objectives:
- This course provides an introduction to the script programming paradigm
- Introduces scripting languages such as Perl, PHP and Python.
- Learning TCL

Course Outcomes:
- Comprehend the differences between typical scripting languages and typical system and application programming languages.
- Gain knowledge of the strengths and weakness of Perl, PHP TCL and Python; and select an appropriate language for solving a given problem

UNIT – I
Introduction to PERL and Scripting, Scripts and Programs, Origin of Scripting , Scripting Today, Characteristics of Scripting Languages, Uses for Scripting Languages, Web Scripting, and the universe of Scripting Languages. PERL- Names and Values, Variables, Scalar Expressions, Control Structures, arrays, list, hashes, strings, pattern and regular expressions, subroutines.

UNIT – II Advanced perl,
Finer points of looping, pack and unpack, filesystem, eval, data structures, packages, modules, objects, interfacing to the operating system, Creating Internet ware applications, Dirty Hands Internet Programming, security Isses.

UNIT – III PHP Basics
Features, Embedding PHP Code in your Web pages, Outputting the data to the browser, Datatypes, Variables, Constants, expressions, string interpolation, control structures . Function, Creating a Function, Function Libraries, Arrays, strings and Regular Expressions.
PHP and Web Forms, Files, PHP Authentication and Methodologies -Hard Coded, File Based, Database Based, IP Based, Login Administration, Uploading Files with PHP, Sending Email using PHP, PHP Encryption Functions, the Mcrypt package, Building Web sites for the World.

UNIT - IV TCL
TCL Structure, syntax, Variables and Data in TCL, Control Flow, Data Structures, input/output, procedures , strings , patterns, files, Advance TCL- eval, source, exec and uplevel commands, Name spaces, trapping errors, event driven programs, making applications internet aware, Nuts and Bolts Internet Programming, Security Issues, C Interface.
Tk
Tk-Visual Tool Kits, Fundamental Concepts of Tk, Tk by example, Events and Binding , Perl-Tk.

UNIT – V Python
Introduction to Python language, python-syntax, statements, functions, Built-in-functions and Methods, Modules in python, Exception Handling.
Integrated Web Applications in Python – Building Small, Efficient Python Web Systems ,Web Application Framework.

TEXT BOOKS:

1. The World of Scripting Languages, David Barron, Wiley Publications.
2. Python Web Programming, Steve Holden and David Beazley, New Riders Publications.
3. Beginning PHP and MySQL , 3rd Edition , Jason Gilmore, Apress Publications (Dream tech.).

REFERENCE BOOKS:

1. Open Source Web Development with LAMP using Linux Apache, MySQL, Perl and PHP, J. Lee and B. Ware (Addison Wesley) Pearson Education.
2. Programming Python,M.Lutz,SPD.
3. PHP 6 Fast and Easy Web Development, Julie Meloni and Matt Telles, Cengage Learning Publications.
4. PHP 5.1,I. Bayross and S. Shah, The X Team, SPD.
5. Core Python Programming, Chun, Pearson Education.
6. Guide to Programming with Python, M. Dawson, Cengage Learning.
7. Perl by Example, E. Quigley, Pearson Education.
8. Programming Perl, Larry Wall, T. Christiansen and J. Orwant, O'Reilly, SPD.
9. Tcl and the Tk Tool kit, Ousterhout, Pearson Education.
10. PHP and MySQL by Example, E. Quigley, Prentice Hall (Pearson).
11. Perl Power, J.P. Flynt, Cengage Learning.
12. PHP Programming solutions, V. Vaswani, TMH.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS I Year II Semester

OPERATIONS RESEARCH

Prerequisites
- A course on "Mathematics"

Course Objectives
- This course explains various Real time problems and Operation Research Techniques to address those problems.
- To study Linear Programming, dynamic programming Operation Research Techniques etc.
- To understand Theory of games.

Course Outcomes
- Gain the knowledge of Operation Research Techniques
- Get the skill to apply Operation Research Techniques to address the real time problems .

UNIT - I
DEVELOPMENT DEFINITION: Characteristics and phases scientific Method, Types of models, General methods for solving. Operations Research models.
ALLOCATION: Introduction, Linear programming Formulation, Graphical solution, Simplex method, artificial variable technique, Duality principle.
TRANSPORTATION PROBLEM: Formulation, optimal solution, un-balanced transportation problem, Degeneracy. Assignment problem: formulation optimal solution, variations. 1.a non-square (mxn) Matrix, Restrictions.

UNIT - II
EEQUE CEING: Introduction, optimal solution for processing each of n-jobs through three machines, travelling salesman problem i.e., shortest acyclic route models.
REPLACEMENT: Introduction, replacement of items that deteriorate when money value is not counted and counted, replacement items that fail completely i.e., group replacements.

UNIT - III
WAITING LINES: Introduction, single channel, poisson arrivals, exponential service times, unrestricted queue, with infinite population and finite population models, single channel, poisson arrivals, exponential service times with infinite population and restricted queue, multi channel, poisson arrivals, exponential service times with infinite population and unrestricted queue.

UNIT - IV
INVENTORY: Introduction, single item deterministic models, production is instantaneous or at a constant rate, shortages are allowed or not allowed and withdrawals from stock is continuous, purchase inventory model with one price break, shortages are not allowed, Instantaneous production demand, production or purchase cost is relevant, stochastic models, demand may be discrete or variable or instantaneous production, instantaneous demand and no setup cost.

UNIT - V
THEORY OF GAMES: Introduction, Minimax (maximum) criterion and optimal strategy, solution of games with saddle points, rectangular games without saddle points.
DYNAMIC PROGRAMMING: Introduction, Billman's Principal of optimality, solution of problems with finite number of stages.

TEXT BOOKS:
1. S. D. Sharma : Operations Research
   kedar Nath Ramnath,1972
2. P. K. Gupta & D. S. Hira : Operations Research
3. R. D. Asrhedkar & R. V. Kulkarni : Operations Research.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS I Year II Semester

SOFTWARE ENGINEERING

Prerequisites
- A course on "Computer Programming and Data Structures"
- A course on "Object Oriented Programming Through Java"

Course Objectives
- The aim of the course is to provide an understanding of the working knowledge of the techniques for estimation, design, testing and quality management of large software development projects.
- Topics include process models, software requirements, software design, software testing, software process/product metrics, risk management, quality management and UML diagrams

Course Outcomes
- Ability to translate end-user requirements into system and software requirements, using e.g. UML, and structure the requirements in a Software Requirements Document (SRD).
- Identify and apply appropriate software architectures and patterns to carry out high level design of a system and be able to critically compare alternative choices.
- Will have experience and/or awareness of testing problems and will be able to develop a simple testing report

UNIT - I:
Introduction to Software Engineering: The evolving role of software, changing nature of software, software myths.
A Generic view of process: Software engineering- a layered technology, a process framework, the capability maturity model integration (CMMI), process patterns, process assessment, personal and team process models.
Process models: The waterfall model, incremental process models, evolutionary process models, the unified process.

UNIT- II:
Software Requirements: Functional and non-functional requirements, user requirements, system requirements, interface specification, the software requirements document.
Requirements engineering process: Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management.
System models: Context models, behavioral models, data models, object models, structured methods.

UNIT- III:
Design Engineering: Design process and design quality, design concepts, the design model.
Creating an architectural design: software architecture, data design, architectural styles and patterns, architectural design, conceptual model of UML, basic structural modeling, class diagrams, sequence diagrams, collaboration diagrams, use case diagrams, component diagrams.

UNIT- IV:
Testing Strategies: A strategic approach to software testing, test strategies for conventional software, black-box and white-box testing, validation testing, system testing, the art of debugging.

Product metrics: Software quality, metrics for analysis model, metrics for design model, metrics for source code, metrics for testing, metrics for maintenance.

UNIT - V:

Metrics for Process and Products: Software measurement, metrics for software quality.

Risk management: Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM, RMMM plan.

Quality Management: Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability, the ISO 9000 quality standards.

TEXT BOOKS:
1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, Mc Graw Hill International Edition.
2. Software Engineering- Sommerville, 7th edition, Pearson Education.
3. The unified modeling language user guide Grady Booch, James Rambaugh, Ivar Jacobson, Pearson Education.

REFERENCES:
1. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley.
2. Software Engineering principles and practice - Waman S Jawadekar, The Mc Graw-Hill Companies.
3. Fundamentals of object oriented design using UML Meiler page-Jones: Pearson Education.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS I Year II Semester

Data Structures Lab Using C++

Prerequisites
- A course on "Computer Programming & Data Structures" and "Advanced Data Base Engineering".

Course Objectives
- To get practical exposure on Advanced Data Structures like AVL Trees, Red-Black trees etc.,
- implementation of data structures such as trees and graphs,
- Programming of sorting and pattern matching algorithms

Course Outcomes
- Ability to select the data structures that efficiently model the information in a problem.
- Ability to assess efficiency trade-offs among different data structure implementations or combinations.
- Implement and know the application of algorithms for sorting and pattern matching.
- Design programs using tree structures, including Optimal binary search tree, All pairs shortest path problem, AVL tree, Red-Black trees.

1. Write a program to implement the following sorting techniques
   a. Merge sort
   b. Quick sort
   c. Bubble sort
   d. Selection sort
   e. Insertion sort
   f. Radix sort
   g. Heap sort
2. Write a program to implement the single source shortest path problem
3. Write a program to implement the Optimal binary search tree
4. Write a program to implement the All pairs shortest path problem
5. Write a program to implement the n-queens problem
6. Write a program to implement the following searching techniques
   a. Linear Search
   b. Binary Search
7. Write a program to implement the operations of Binary search tree
8. Write a program to implement the tree traversal methods
9. Write a program to implement the graph traversal methods
10. Write a program to implement the operations of AVL tree
11. Write a program to implement the operations of Red-Black tree
12. Write a program to implement the following Pattern matching algorithms
    a. Brute Force algorithm
    b. Boyer Moore algorithm
    c. Knuth- Morris-Pratt algorithm

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS I Year II Semester

SCRIPTING (PYTHON) PROGRAMMING LAB

Course Objectives:
- To be able to introduce core programming basics and program design with functions using Python programming language.
- To understand a range of Object-Oriented Programming, as well as in-depth data and information processing techniques.
- To understand the high-performance programs designed to strengthen the practical expertise.

Course Outcomes:
1. Student should be able to understand the basic concepts scripting and the contributions of scripting language
2. Ability to explore python especially the object-oriented concepts, and the built-in objects of Python.
3. Ability to create practical and contemporary applications such as TCP/IP network programming, Web applications, discrete-event simulations

Prerequisites:
- Students should install Python on Linux platform.

List of Programs:
1. Write a program to demonstrate different number data types in Python.
2. Write a program to perform different Arithmetic Operations on numbers in Python.
3. Write a program to create, concatenate and print a string and accessing sub-string from a given string.
4. Write a python script to print the current date in the following format "Sun May 29 02:26:23 IST 2017"
5. Write a program to create, append, and remove lists in python.
6. Write a program to demonstrate working with tuples in python.
7. Write a program to demonstrate working with dictionaries in python.
8. Write a python program to find largest of three numbers.
9. Write a Python program to convert temperatures to and from Celsius, Fahrenheit. [Formula : c/5 = f-32/9]
10. Write a Python program to construct the following pattern, using a nested for loop

```
        *
    *                                                                    *
    *                                *                                    *
    *                *                *                *                  *
    *            *        *              *              *            *    *
    *                *                *                *                  *
    *                                *                                    *
    *                                                                    *
    *
```

11. Write a Python script that prints prime numbers less than 20.
12. Write a python program to find factorial of a number using Recursion.
13. Write a program that accepts the lengths of three sides of a triangle as inputs. The program output should indicate whether or not the triangle is a right triangle (Recall from the

Pythagorean Theorem that in a right triangle, the square of one side equals the sum of the squares of the other two sides).

14. Write a python program to define a module to find Fibonacci Numbers and import the module to another program.
15. Write a python program to define a module and import a specific function in that module to another program.
16. Write a script named **copyfile.py**. This script should prompt the user for the names of two text files. The contents of the first file should be input and written to the second file.
17. Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order.
18. Write a Python class to convert an integer to a roman numeral.
19. Write a Python class to implement pow(x, n)
20. Write a Python class to reverse a string word by word.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS I Year II Semester

English Language Communication Skills Lab

Prerequisite: NIL

Course Objectives
- To facilitate computer-aided multi-media instruction enabling individualized and independent language learning
- To sensitise the students to the nuances of English speech sounds, word accent, intonation and rhythm
- To bring about a consistent accent and intelligibility in their pronunciation of English by providing an opportunity for practice in speaking
- To improve the fluency in spoken English and neutralize mother tongue influence
- To train students to use language appropriately for interviews, group discussion and public speaking

Learning Outcomes
- Better Understanding of nuances of language through audio- visual experience and group activities
- Neutralization of accent for intelligibility
- Speaking with clarity and confidence thereby enhancing employability skills of the students

Syllabus: English Language Communication Skills Lab shall have two parts:
- a. **Computer Assisted Language Learning (CALL) Lab**
- b. **Interactive Communication Skills (ICS) Lab**

The following course content is prescribed for the **English Language Communication Skills Lab**
Exercise – I
CALL Lab: Introduction to Phonetics – Speech Sounds – Vowels and Consonants
ICS Lab: Ice-Breaking activity and JAM session
Articles, Prepositions, Word formation- Prefixes & Suffixes, Synonyms & Antonyms

Exercise – II
CALL Lab: Structure of Syllables - Past Tense Marker and Plural Marker – Weak Forms and Strong Forms - Consonant Clusters.
ICS Lab: Situational Dialogues – Role-Play- Expressions in Various Situations – Self-introduction and Introducing Others – Greetings – Apologies – Requests – Social and Professional Etiquette - Telephone Etiquette.
Concord (Subject in agreement with verb) and Words often misspelt- confused/misused

Exercise - III
CALL Lab: Minimal Pairs- Word accent and Stress Shifts- Listening Comprehension.
ICS Lab: Descriptions- Narrations- Giving Directions and guidelines.
Sequence of Tenses, Question Tags and One word substitutes.

Exercise – IV
CALL Lab: Intonation and Common errors in Pronunciation.
ICS Lab: Extempore- Public Speaking
Active and Passive Voice, –Common Errors in English, Idioms and Phrases

Exercise – V
CALL Lab: Neutralization of Mother Tongue Influence and Conversation Practice
ICS Lab: Information Transfer- Oral Presentation Skills
Reading Comprehension and Job Application with Resume preparation.

Minimum Requirement of infrastructural facilities for ELCS Lab:
    1. Computer Assisted Language Learning (CALL) Lab:
The Computer aided Language Lab for 40 students with 40 systems, one master console, LAN facility and English language software for self- study by learners.

System Requirement (Hardware component):
Computer network with Lan with minimum 60 multimedia systems with the following specifications:
    i)   P – IV Processor
        a) Speed – 2.8 GHZ
        b) RAM – 512 MB Minimum
        c) Hard Disk – 80 GB
    ii)   Headphones of High quality

    2. Interactive Communication Skills (ICS) Lab :
The Interactive Communication Skills Lab: A Spacious room with movable chairs and audio-visual aids with a Public Address System, a T. V., a digital stereo –audio & video system and camcorder etc.

Suggested Software:
1. **Cambridge Advanced Learners' English Dictionary with CD.**
2. **Grammar Made Easy by Darling Kindersley**
3. **Punctuation Made Easy by Darling Kindersley**
4. Clarity Pronunciation Power – Part I
5. Clarity Pronunciation Power – part II
6. **Oxford Advanced Learner's Compass, 8th Edition**
7. *DELTA's key to the Next Generation TOEFL Test: Advanced Skill Practice.*
8. Lingua TOEFL CBT Insider, by Dreamtech
9. TOEFL & GRE (KAPLAN, AARCO & BARRONS, USA, Cracking GRE by CLIFFS)
10. **English in Mind (Series 1-4), Herbert Puchta and Jeff Stranks with Meredith Levy, Cambridge**
11. **English Pronunciation in Use** (Elementary, Intermediate, Advanced) Cambridge University Press
12. Raman, M & Sharma, S. 2011. Technical Communication, OUP
13. Sanjay Kumar & Pushp Lata. 2011. Communication Skills, OUP

SUGGESTED READING:
1. Rama Krishna Rao, A. *et al. English Language Communication Skills – A Reader cum Lab Manual Course Content and Practice.* Chennai: Anuradha Publishers
2. Suresh Kumar, E. & Sreehari, P. 2009. *A Handbook for English Language Laboratories.* New Delhi: Foundation
3. *Speaking English Effectively* 2nd Edition by Krishna Mohan and N. P. Singh, 2011. Macmillan Publishers India Ltd. Delhi.
4. Sasi Kumar, V & Dhamija, P.V. *How to Prepare for Group Discussion and Interviews.* Tata McGraw Hill
5. Spoken English: A Manual of Speech and Phonetics by R. K. Bansal & J. B. Harrison. 2013. Orient Blackswan. Hyderabad.
6. *English Pronunciation in Use. (Elementary, Intermediate & Advance).* Cambridge: CUP

7. Chris Redston, Gillie Cunningham, Jan Bell. *Face to Face* (2nd Edition). Cambridge University Press
8. Nambiar, K.C. 2011. *Speaking Accurately. A Course in International Communication.* New Delhi : Foundation
9. Soundararaj, Francis. 2012. *Basics of Communication in English. New Delhi: Macmillan*
10. A textbook of English Phonetics for Indian Students by T. Balasubramanian (Macmillan)

DISTRIBUTION AND WEIGHTAGE OF MARKS

English Language Laboratory Practical Examination:
1. The practical examinations for the English Language Laboratory shall be conducted as per the University norms prescribed for the core engineering practical sessions.
2. For the Language lab sessions, there shall be a continuous evaluation during the year for 30 sessional marks and 70 semester-end Examination marks. Of the 30 marks, 20 marks shall be awarded for day-to-day work and 10 marks to be awarded by conducting Internal Lab Test(s). The year- end Examination shall be conducted by the teacher concerned with the help of another member of the staff of the same department of the same institution.

* * * * *

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS II Year I Semester

DATABASE MANAGEMENT SYSTEMS

Prerequisites
- A course on "Advanced Data Structures"

Course Objectives
- To understand the basic concepts and the applications of database systems.
- To master the basics of SQL and construct queries using SQL.
- Topics include data models, database design, relational model, relational algebra, transaction control, concurrency control, storage structures and access techniques.

Course Outcomes
- Gain knowledge of fundamentals of DBMS, database design and normal forms
- Master the basics of SQL for retrieval and management of data.
- Be acquainted with the basics of transaction processing and concurrency control.
- Familiarity with database storage structures and access techniques

UNIT - I:
Database System Applications: database system Vs. file system, view of data, data abstraction, instances and schemas, data models, the ER model, relational model, other models, database languages, DDL, DML, database access for application programs, database users and administrator, transaction management, database system structure, storage manager, the query processor, history of data base systems, data base design and ER diagrams, beyond ER design entities, attributes and entity sets, relationships and relationship sets, additional features of ER model, concept design with the ER Model, conceptual design for large enterprises.

UNIT - II:
Introduction to the Relational Model: integrity constraint over relations, enforcing integrity constraints, querying relational data, logical data base design, introduction to views, destroying/altering tables and views, form of basic SQL query, examples of basic SQL queries, introduction to nested queries, correlated nested queries, set comparison operators, aggregation operators, NULL values, comparison using null values, logical connectivity's, AND, OR and NOT, impact on SQL constructs, outer joins, disallowing NULL values, complex integrity constraints in SQL, triggers and active data bases, Oracle, SQL Server, DB2.

UNIT - III:
Relational Algebra: Selection and projection, set operations, renaming, Joins, Division, Examples of Algebra overviews, Relational calculus, Tuple relational Calculus, Domain relational calculus, Expressive Power of Algebra and calculus.
Schema refinement: Problems caused by redundancy, decompositions, problems related to decomposition, reasoning about functional dependencies, FIRST, SECOND, THIRD normal forms, BCNF, lossless join decomposition, dependency preserving decomposition, schema refinement in database design, multi valued dependencies, FOURTH normal form, FIFTH normal form.

UNIT - IV:
Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for serializability, Lock Based Protocols, Timestamp Based Protocols, Validation- Based Protocols, Multiple Granularity.

Recovery and Atomicity, Log–Based Recovery, Recovery with Concurrent Transactions, Buffer Management, Failure with loss of nonvolatile storage, Advance Recovery systems, Remote Backup systems.

UNIT - V:
Data on External Storage, File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes, Index data Structures, Hash Based Indexing, Tree base Indexing, Comparison of File Organizations, Indexes and Performance Tuning, Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM), B+ Trees: A Dynamic Index Structure.

TEXT BOOKS:
1. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata Mc Graw Hill 3rd Edition
2. Database System Concepts, Silberschatz, Korth, Mc Graw hill, V edition.

REFERENCES:
1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navrate Pearson Education
3. Introduction to Database Systems, C. J. Date Pearson Education
4. Oracle for Professionals, The X Team, S. Shah and V. Shah, SPD.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. Fundamentals of Database Management Systems, M. L. Gillenson, Wiley Student Edition.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS II Year I Semester

COMPUTER NETWORKS

Prerequisites
- A course on "Computer Programming and Data Structures"
- A course on "Design and Analysis of Algorithms"

Course Objectives
- The objective of the course is to equip the students with a general overview of the concepts and fundamentals of computer networks.
- Familiarize the students with the standard models for the layered approach to communication between machines in a network and the protocols of the various layers.
- The following topics are included: Reference models, the physical layer (transmission media); the data link layer (error detection and correction, point-to-point protocols); the medium access layer protocols; the network layer (routing algorithms, congestion control); internetworking (addressing, internetwork routing and protocols, quality of service); the transport layer (connection-oriented transport layer services and protocols); application layer protocols

Course Outcomes
- Gain the knowledge of the basic computer network technology.
- Gain the knowledge of the functions of each layer in the OSI and TCP/IP reference model.
- Obtain the skills of subnetting and routing mechanisms.
- Familiarity with the essential protocols of computer networks, and how they can be appled in network design and implementation.

UNIT- I
Network hardware, Network software, OSI, TCP/IP Reference models, Example Networks: ARPANET, Internet.
Physical Layer: Guided Transmission media: twisted pairs, coaxial cable, fiber optics, Wireless transmission.

UNIT - II
Data link layer: Design issues, framing, Error detection and correction.
Elementary data link protocols: simplex protocol, A simplex stop and wait protocol for an error-free channel, A simplex stop and wait protocol for noisy channel.
Sliding Window protocols: A one-bit sliding window protocol, A protocol using Go-Back-N, A protocol using Selective Repeat, Example data link protocols.
Medium Access sub layer: The channel allocation problem, Multiple access protocols: ALOHA, Carrier sense multiple access protocols, collision free protocols. Wireless LANs, Data link layer switching.

UNIT- III
Network Layer: Design issues, Routing algorithms: shortest path routing, Flooding, Hierarchical routing, Broadcast, Multicast, distance vector routing, Congestion Control Algorithms, Quality of Service, Internetworking, The Network layer in the internet.

UNIT – IV
Transport Layer: Transport Services, Elements of Transport protocols, Connection management, TCP and UDP protocols.

UNIT – V

Application Layer –Domain name system, SNMP, Electronic Mail; the World WEB, HTTP, Streaming audio and video.

TEXT BOOK:

1. Computer Networks -- Andrew S Tanenbaum, David. j. Wetherall, 5th Edition. Pearson Education/PHI

REFERENCE BOOKS:

1. An Engineering Approach to Computer Networks - S. Keshav, 2nd Edition, Pearson Education
2. Data Communications and Networking – Behrouz A. Forouzan. Third Edition TMH.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS II Year I Semester

OPERATING SYSTEMS

Prerequisites:
- A course on "Computer Programming and Data Structures"
- A course on "Computer Organization and Architecture"

Course Objectives:
- Provide an introduction to operating system concepts (i.e., processes, threads, scheduling, synchronization, deadlocks, memory management, file and I/O subsystems and protection)
- Introduce the issues to be considered in the design and development of operating system
- Introduce basic Unix commands, system call interface for process management, interprocess communication and I/O in Unix

Course Outcomes:
- Will be able to control access to a computer and the files that may be shared
- Demonstrate the knowledge of the components of computer and their respective roles in computing.
- Ability to recognize and resolve user problems with standard operating environments.
- Gain practical knowledge of how programming languages, operating systems, and architectures interact and how to use each effectively.

UNIT - I:
Operating System Introduction, Structures - Simple Batch, Multiprogrammed, Time-shared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, System components, Operating System services, System Calls, Virtual Machines, System Design and Implementation.
UNIX/LINUX Utilities - Introduction to Unix file system, vi editor, file handling utilities, security by file permissions, process utilities, disk utilities, networking commands, text processing utilities and backup utilities.

UNIT - II:
Process and CPU Scheduling - Process concepts and scheduling, Operations on processes, Cooperating Processes, Threads, and Interposes Communication, Scheduling Criteria, Scheduling Algorithms, Multiple -Processor Scheduling, Real-Time Scheduling.
System call interface for process management-fork, exit, wait, waitpid, exec
Deadlocks - System Model, Deadlocks Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock.

UNIT – III:
Process Management and Synchronization - The Critical Section Problem, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Critical Regions, Monitors.
Interprocess Communication Mechanisms: IPC between processes on a single computer system, IPC between processes on different systems, using pipes, FIFOs, message queues, shared memory, semaphores.

UNIT - IV
Memory Management and Virtual Memory - Logical versus Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging, Demand Paging, Performance of Demanding Paging, Page Replacement, Page Replacement Algorithms, Allocation of Frames, Thrashing.

UNIT - V:

File System Interface and Implementation -Access methods, Directory Structure, Protection, File System Structure, Allocation methods, Free-space Management, Directory Management, Directory Implementation, Efficiency, and Performance.

Unix/LINUX Files: File structure, directories, files and devices, System calls, library functions, low level file access, usage of open, create, read, write, close, lseek, stat, ioctl.

TEXT BOOKS:

1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, John Wiley
2. Advanced programming in the Unix environment, W. R. Stevens, Pearson education.

REFERENCE BOOKS:

1. Operating Systems – Internals and Design Principles Stallings, Fifth Edition–2005, Pearson Education/PHI
2. Operating System A Design Approach-Crowley, TMH.
3. Modern Operating Systems, Andrew S Tanenbaum 2nd edition, Pearson/PHI
4. Unix programming environment, Kernighan and Pike, PHI. / Pearson Education
5. Unix Internals The New Frontiers, U. Vahalia, Pearson Education.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS II Year I Semester

OBJECT ORIENTED ANALYSIS AND DESIGN USING UML

Prerequisites:
- Software Engineering

Course Objectives:
- To train students on object modeling
- To apply unified process phases
- To apply unified modeling language for software design of any applications
- To study case studies for OOAD

Course Outcomes:
- Will be able to use UML notations
- Can apply unified process in software development
- Will be able to perform analysis and design using object modeling

UNIT - I
Introduction to UML: Importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture, Software Development Life Cycle.

UNIT - II
Basic Structural Modeling: Classes, Relationships, common Mechanisms, and diagrams.
Advanced Structural Modeling: Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages.
Class & Object Diagrams: Terms, concepts, modeling techniques for Class & Object   Diagrams.

UNIT - III
Basic Behavioral Modeling-I: Interactions, Interaction diagrams.
Basic Behavioral Modeling-II: Use cases, Use case Diagrams, Activity Diagrams.

UNIT - IV
Advanced Behavioral Modeling: Events and signals, state machines, processes and Threads, time and space, state chart diagrams.
Architectural Modeling: Component, Deployment, Component diagrams and Deployment diagrams.

UNIT - V
Patterns and Frameworks, Artifact Diagrams. Case Study: The Unified Library application

TEXT BOOKS:
1. Grady Booch, James Rumbaugh, Ivar Jacobson : The Unified Modeling Language User Guide, Pearson Education 2nd Edition
2. Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado: UML 2 Toolkit, WILEY-Dreamtech India Pvt. Ltd.

REFERENCE BOOKS:
1. Meilir Page-Jones: Fundamentals of Object Oriented Design in UML, Pearson Education.
2. Pascal Roques: Modeling Software Systems Using UML2, WILEY-Dreamtech India Pvt. Ltd.
3. Atul Kahate: Object Oriented Analysis & Design, The McGraw-Hill Companies.
4. Mark Priestley: Practical Object-Oriented Design with UML,TMH.

5. Appling UML and Patterns: An introduction to Object – Oriented Analysis and Design and Unified Process, Craig Larman, Pearson Education.
6. Object-Oriented Analysis and Design with the Unified Process By John W. Satzinger, Robert B Jackson and Stephen D Burd, Cengage Learning.
7. UML and C++,R. C. Lee, and W. M. Tepfenhart, PHI.
8. Object Oriented Analysis, Design and Implementation, B. Dathan, S. Ramnath, Universities Press.
9. OODesign with UML and Java, K. Barclay, J. Savage, Elsevier.
10. Learning UML 2.0, Russ Miles and Kim Hamilton, O'riley, SPD.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS II Year I Semester

JAVA PROGRAMMING

Prerequisites
- A course on "Computer Programming & Data Structures"

Course Objectives
- Introduces object oriented programming concepts using the Java language.
- Introduces the principles of inheritance and polymorphism; and demonstrates how they relate to the design of abstract classes
- Introduces the implementation of packages and interfaces
- Introduces exception handling, event handling and multithreading
- Introduces the design of Graphical User Interface using applets and swings

Course Outcomes
- Develop applications for a range of problems using object-oriented programming techniques
- Design simple Graphical User Interface applications

UNIT - I :
Object oriented thinking and Java Basics- Need for oop paradigm, summary of oop concepts, coping with complexity, abstraction mechanisms. A way of viewing world – Agents, responsibility, messages, methods, History of Java, Java buzzwords, data types, variables, scope and life time of variables, arrays, operators, expressions, control statements, type conversion and casting, simple java program, concepts of classes, objects, constructors, methods, access control, this keyword, garbage collection, overloading methods and constructors, method binding, inheritance, overriding and exceptions, parameter passing, recursion, nested and inner classes, exploring string class.

UNIT - II :
Inheritance, Packages and Interfaces – Hierarchical abstractions, Base class object, subclass, subtype, substitutability, forms of inheritance- specialization, specification, construction, extension, limitation, combination, benefits of inheritance, costs of inheritance. Member access rules, super uses, using final with inheritance, polymorphism- method overriding, abstract classes, the Object class.
Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages, differences between classes and interfaces, defining an interface, implementing interface, applying interfaces, variables in interface and extending interfaces.
Exploring java.io.

UNIT - III
Exception handling and Multithreading-- Concepts of exception handling, benefits of exception handling, Termination or resumptive models, exception hierarchy, usage of try, catch, throw, throws and finally, built in exceptions, creating own exception sub classes.
String handling,Exploring java.util.Differences between multi threading and multitasking, thread life cycle, creating threads, thread priorities, synchronizing threads, interthread communication, thread groups, daemon threads.
Enumerations, autoboxing, annotations, generics.

UNIT - IV :
Event Handling : Events, Event sources, Event classes, Event Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes.

The AWT class hierarchy, user interface components- labels, button, canvas, scrollbars, text components, check box, check box groups, choices, lists panels – scrollpane, dialogs, menubar, graphics, layout manager – layout manager types – border, grid, flow, card and grid bag.

UNIT - V:
Applets – Concepts of Applets, differences between applets and applications, life cycle of an applet, types of applets, creating applets, passing parameters to applets.
Swing – Introduction, limitations of AWT, MVC architecture, components, containers, exploring swing- JApplet, JFrame and JComponent, Icons and Labels, text fields, buttons – The JButton class, Check boxes, Radio buttons, Combo boxes, Tabbed Panes, Scroll Panes, Trees, and Tables.

TEXT BOOKS:
1. Java the complete reference, 7th edition, Herbert Schildt, TMH.
2. Understanding OOP with Java, updated edition, T. Budd, Pearson Education.

REFERENCES:
1. An Introduction to programming and OO design using Java, J. Nino and F.A. Hosch, John Wiley & sons.
2. Introduction to Java programming, Y. Daniel Liang, Pearson Education.
3. An introduction to Java programming and object oriented application development, R.A. Johnson- Thomson.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS II Year I Semester

DATABASE MANAGEMENT SYSTEMS LAB

Pre-requisites:
- Co-requisite of course "Database Management Systems"

Course Objectives:
- Introduce ER data model, database design and normalization
- Learn SQL basics for data definition and data manipulation

Course Outcomes:
- Design database schema for a given application and apply normalization
- Acquire skills in using SQL commands for data definition and data manipulation.
- Develop solutions for database applications using procedures, cursors and triggers

List of Experiments:-
1) Concept design with E-R Model
2) Relational Model
3) Normalization
4) Practicing DDL commands
5) Practicing DML commands
6) Querying (using ANY, ALL, IN, Exists, NOT EXISTS, UNION, INTERSECT, Constraints etc.)
7) Queries using Aggregate functions, GROUP BY, HAVING and Creation and dropping of Views.
8) Triggers (Creation of insert trigger, delete trigger, update trigger)
9) Procedures
10) Usage of Cursors

TEXT BOOKS:
1. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata Mc Graw Hill 3rd Edition
2. Database System Concepts, Silberschatz, Korth, Mc Graw hill, V edition.

REFERENCES:
1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navrate Pearson Education
3. Introduction to Database Systems, C. J. Date Pearson Education
4. Oracle for Professionals, The X Team, S. Shah and V. Shah, SPD.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. Fundamentals of Database Management Systems, M. L. Gillenson, Wiley Student Edition.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS II Year I Semester

JAVA PROGRAMMING LAB

Prerequisites
- A course on "Computer Programming & Data Structures"

Co-requisite
- A Course on "Object-Oriented Programming Through Java"

Course Objectives
- Introduces object oriented programming concepts using the Java language.
- Introduces the principles of inheritance and polymorphism; and demonstrates how they relate to the design of abstract classes
- Introduces the implementation of packages and interfaces
- Introduces exception handling, event handling and multithreading
- Introduces the design of Graphical User Interface using applets and swings

Course Outcomes
- Develop applications for a range of problems using object-oriented programming techniques
- Design simple Graphical User Interface applications

Use Eclipse or Netbean platform and get acquainted with the various menus. Create a test project, add a test class and run it. See how you can use auto suggestions, auto fill. Try code formatter and code refactoring like renaming variables, methods and classes. Try debug step by step with a small program of about 10 to 15 lines which contains at least one if else condition and a for loop.

1) Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the exception in a message dialog box. [Use JOption Pane – Input dialog, Message dialog]

2) Write a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method printArea () that prints the area of the given shape.

3) Write a Java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.

4) Write a Java program that connects to a database using JDBC and does add, delete, modify and retrieve operations.

5) Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with "Stop" or "Ready" or "Go" should appear above the buttons in selected color. Initially, there is no message shown.

6) Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -, *, % operations. Add a text field to display the result. Handle any possible exceptions like divide by zero

7) a) Develop an applet in Java that displays a simple message.
   b) Develop an applet in Java that receives an integer in one text field, and computes its factorial value and returns it in another text field, when the button named "Compute" is clicked.

8) Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a Java program to display the table using Labels in Grid Layout.

9) Write a Java program that loads names and phone numbers from a text file where the data is organized as one line per record and each field in a record are separated by a tab (\t). It takes a name or phone number as input and prints the corresponding other value from the hash table (hint: use hash tables).

10) Implement the above program with the database instead of a text file.

11) Write a Java program that prints the meta-data of a given table

TEXT BOOK:
   1. Java Fundamentals – A comprehensive Introduction, Herbert Schildt and Dale Skrien, TMH.

REFERENCES:
   1. Java for Programmers, P. J. Deitel and H. M. Deitel, Pearson education (OR) Java: How to Program P. J. Deitel and H. M. Deitel, PHI.
   2. Object Oriented Programming through Java, P. Radha Krishna, Universities Press.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS II Year II Semester

WEB TECHNOLOGIES

Course Objectives:
- To introduce PHP language for server side scripting
- To introduce XML and processing of XML Data with Java
- To introduce Server side programming with Java Servlets and JSP
- To introduce Client side scripting with Javascript and AJAX.

Course Outcomes: The Student is expected to
- gain knowledge of client side scripting, validation of forms and AJAX programming
- have understanding of server side scripting with PHP language
- have understanding of what is XML and how to parse and use XML Data with Java
- To introduce Server side programming with Java Servlets and JSP

UNIT- I
Introduction to PHP: Declaring variables, data types, arrays, strings, operators, expressions, control structures, functions, Reading data from web form controls like text boxes, radio buttons, lists etc., Handling File Uploads, Connecting to database (MySQL as reference), executing simple queries, handling results, Handling sessions and cookies
File Handling in PHP: File operations like opening, closing, reading, writing, appending, deleting etc. on text and binary files, listing directories

UNIT- II
XML: Introduction to XML, Defining XML tags, their attributes and values, Document Type Definition, XML Schemas, Document Object Model, XHTML
Parsing XML Data - DOM and SAX Parsers in java.

UNIT- III
Introduction to Servlets: Common Gateway Interface (CGI), Lifecycle of a Servlet, deploying a servlet, The Servlet API, Reading Servlet parameters, Reading Initialization parameters, Handling Http Request & Responses, Using Cookies and Sessions, connecting to a database using JDBC.

UNIT- IV
Introduction to JSP: The Anatomy of a JSP Page, JSP Processing, Declarations, Directives, Expressions, Code Snippets, implicit objects, Using Beans in JSP Pages, Using Cookies and session for session tracking, connecting to database in JSP.

UNIT- V
Client side Scripting: Introduction to Javascript: Javascript language - declaring variables, scope of variables, functions, event handlers (onclick, onsubmit etc.), Document Object Model, Form validation. Simple AJAX application.

TEXT BOOKS:
1. Web Technologies, Uttam K Roy, Oxford University Press
2. The Complete Reference PHP – Steven Holzner, Tata McGraw-Hill

REFERENCES:
1. Web Programming, building internet applications, Chris Bates 2nd edition, Wiley Dreamtech
2. Java Server Pages –Hans Bergsten, SPD O'Reilly

3. Java Script, D. Flanagan, O'Reilly, SPD.
4. Beginning Web Programming-Jon Duckett WROX.
5. Programming World Wide Web, R. W. Sebesta, Fourth Edition, Pearson.
6. Internet and World Wide Web – How to program, Dietel and Nieto, Pearson.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS II Year II Semester

Linux Programming

UNIT – I
Problem solving approaches in UNIX:
Using single commands, using compound Commands, shell scripts, C programs, building own command library of programs. **Working with the Bourne shell:** Introduction, shell responsibilities, pipes and input Redirection, output redirection, here documents, running a shell script, the shell as a programming language, shell meta characters, file name substitution, shell variables, command substitution, shell commands, the environment, quoting, test command, control structures, arithmetic in shell, shell script examples, interrupt processing, functions, debugging shell scripts.

UNIT - II
Unix Files: File Concept, File Structure, File System Layout, File types, The standard I/O (fopen, fclose, fflush, fseek, fgetc, getc, getchar, fputc, putc, putchar, fgets, gets etc.), formatted I/O, stream errors, kernel support for files, System calls, library functions, file descriptors, low level file access - usage of open, creat, read, write, close, lseek, stat family, umask, dup, dup2, fcntl. file and directory management - chmod, chown, links(soft links & hard links - unlink, link, symlink), mkdir, rmdir, chdir, getcwd, opendir, readdir, closedir, rewinddir, seekdir, telldir functions.
Basics of Perl - Scalars and their operations, assignment statements and simple I/O, control statements, arrays, hashes, references, functions, pattern matching, file I/O, example programs.

UNIT - III
Unix Process – Process concept, Kernel support for process, process attributes, process hierarchy, process creation, waiting for a process, process termination, process control, zombie process, **system call interface for process management**-fork, vfork, exit, wait, waitpid, exec family, system.
 Unix Signals – Introduction to signals, Signal generation and handling, Kernel support for signals, Signal function, unreliable signals, reliable signals, kill, raise , alarm, pause, abort, sleep functions.

UNIT – IV
Interprocess Communication Overview: Introduction to IPC, IPC between processes on a single computer system, IPC between processes on different systems, file and record locking, other unix locking techniques, pipes, FIFOs, namespaces, introduction to three types of IPC(system-V)-message queues, semaphores and shared memory.
Message Queues-Unix system-V messages, unix kernel support for messages, unix APIs for messages, client/server example.

UNIT - V:
Semaphores-Unix system-V semaphores, unix kernel support for semaphores, unix APIs for semaphores, file locking with semaphores.
Shared Memory-Unix system-V shared memory, unix kernel support for shared memory, unix APIs for shared memory, semaphore and shared memory example.
Sockets: Berkeley sockets, socket system calls for connection oriented protocol and connectionless protocol, example-client/server programs.

TEXT BOOKS:
1. Unix Network Programming, W. R. Stevens, Pearson/PHI.
2. Unix Concepts and Applications, 4th Edition, Sumitabha Das, TMH.
3. Unix system programming using C++, T. Chan, PHI.

REFERENCE BOOKS:

1. Linux System Programming, Robert Love, O'Reilly, SPD.
2. Unix for programmers and users, 3rd Edition, Graham Glass, King Ables, Pearson Education.
3. Advanced Programming in the Unix environment, 2nd Edition, W. R. Stevens & S. A. Rago, Pearson Education.
4. Learning Perl, R. L. Schwartz, T. Phoenix, B.D. Foy, O'Reilly, SPD.
5. Unix Programming, Kumar Saurabh, 1st Edition, Wiley India Pvt Ltd.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS II Year II Semester

Machine Learning (Core Elective – I)

Prerequisites:
- Students are expected to have knowledge in linear signals and systems, Fourier Transform, basic linear algebra, basic probability theory and basic programming techniques; knowledge of Digital Signal Processing is desirable.
- A course on "Computational Mathematics"
- A course on "Computer Oriented Statistical Methods"

Course Objectives:
- This course introduces fundamental concepts, theories, and algorithms for pattern recognition and machine learning.
- Topics include: Pattern Representation, Nearest Neighbour Based Classifier, Bayes Classifier, Hidden Markov Models, Decision Trees, Support Vector Machines, Clustering, and an application of hand-written digit recognition.

Course Outcomes:
- Understand the theory, benefits, inadequacies and possible applications of various machine learning and pattern recognition algorithms
- Identify and employ suitable machine learning techniques in classification, pattern recognition, clustering and decision problems.

UNIT - I
Introduction - Well-posed learning problems, designing a learning system Perspectives and issues in machine learning
Concept learning and the general to specific ordering – Introduction, A concept learning task, concept learning as search, Find-S: Finding a Maximally Specific Hypothesis, Version Spaces and the Candidate Elimination algorithm, Remarks on Version Spaces and Candidate Elimination, Inductive Bias.
Decision Tree Learning – Introduction, Decision Tree Representation, Appropriate Problems for Decision Tree Learning, The Basic Decision Tree Learning Algorithm Hypothesis Space Search in Decision Tree Learning, Inductive Bias in Decision Tree Learning, Issues in Decision Tree Learning.

UNIT - II
Artificial Neural Networks Introduction, Neural Network Representation, Appropriate Problems for Neural Network Learning, Perceptions, Multilayer Networks and the Back propagation Algorithm. Discussion on the Back Propagation Algorithm, An illustrative Example: Face Recognition
Evaluation Hypotheses – Motivation, Estimation Hypothesis Accuracy, Basics of Sampling Theory, A General Approach for Deriving Confidence Intervals, Difference in Error of Two Hypotheses, Comparing Learning Algorithms.

UNIT - III
Bayesian learning - Introduction, Bayes Theorem, Bayes Theorem and Concept Learning Maximum Likelihood and Least Squared Error Hypotheses, Maximum Likelihood Hypotheses for Predicting Probabilities, Minimum Description Length Principle , Bayes Optimal Classifier, Gibs Algorithm, Naïve Bayes Classifier, An Example: Learning to Classify Text, Bayesian Belief Networks, EM Algorithm.
Computational Learning Theory – Introduction, Probably Learning an Approximately Correct Hypothesis, Sample Complexity for Finite Hypothesis Space, Sample Complexity for Infinite Hypothesis Spaces, The Mistake Bound Model of Learning.

Instance-Based Learning – Introduction, k-Nearest Neighbor Learning, Locally Weighted Regression, Radial Basis Functions, Case-Based Reasoning, Remarks on Lazy and Eager Learning.

UNIT - IV
Pattern Comparison Techniques, Temporal patterns, Dynamic Time Warping Methods, Clustering, Codebook Generation, Vector Quantization
Pattern Classification: Introduction to HMMS, Training and Testing of Discrete Hidden Markov Models and Continuous Hidden Markov Models, Viterbi Algorithm, Different Case Studies in Speech recognition and Image Processing

UNIT - V
Analytical Learning – Introduction, Learning with Perfect Domain Theories: PROLOG-EBG Remarks on Explanation-Based Learning, Explanation-Based Learning of Search Control Knowledge, Using Prior Knowledge to Alter the Search Objective, Using Prior Knowledge to Augment Search Operations.
Combining Inductive and Analytical Learning – Motivation, Inductive-Analytical Approaches to Learning, Using Prior Knowledge to Initialize the Hypothesis.

TEXT BOOKS:
1. Machine Learning – Tom M. Mitchell, MGH
2. Fundamentals of Speech Recognition By Lawrence Rabiner and Biing – Hwang Juang.

REFERENCE BOOKS:
1. Machine Learning : An Algorithmic Perspective, Stephen Marsland, Taylor & Francis

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS II Year II Semester

DATA WAREHOUSING AND DATAMINING (Core Elective – I)

Prerequisites:
- A course on "Database Management Systems"
- Knowledge of probability and statistics

Course Objectives:
- This course presents the techniques for preprocessing data before mining, and describes the concepts related to data warehousing, On-Line Analytical Processing (OLAP), and data generalization.
- It also presents methods for mining frequent patterns, associations, and correlations.
- It then describes methods for data classification and prediction, and data-clustering approaches.

Course Outcomes:
- Examine the types of the data to be mined and present a general classification of tasks and primitives to integrate a data mining system.
- Apply preprocessing statistical methods for any given raw data.
- Devise efficient and cost effective methods for designing and maintaining data warehouses.
- Extract interesting patterns from large amounts of data that can be used for further analysis, for example in machine learning and prediction.
- Discover the role played by data mining in various fields.
- Choose and employ suitable data mining algorithms to build analytical applications
- Evaluate the accuracy of supervised and unsupervised models and algorithms.

UNIT- I
DATA MINING
Data-Types of Data-, Data Mining Functionalities- Interestingness Patterns-Classification of Data Mining systems- Data mining Task primitives -Integration of Data mining system with a Data warehouse-Major issues in Data Mining-Data Preprocessing.

UNIT - II
DATA WAREHOUSE AND BUSINESS ANALYSIS
Data Warehouse-Data Warehouse Architecture- Multidimensional Data Model-Data cube and OLAP Technology-Data Warehouse Implementation -DBMS schemas for Decision support - Efficient methods for Data cube computation.

UNIT- III
ASSOCIATION RULE MINING AND CLASSIFICATION
Mining Frequent Patterns-Associations and correlations- Mining Methods- Mining Various kinds of Association Rules- Correlation Analysis- Constraint based Association mining.-Classification and Prediction- Basic concepts-Decision tree induction-Bayesian classification, Rule-based classification - classification by Back propagation,-Support vector machines-.Associative Classification, Lazy learners-Other classification methods – Prediction.

UNIT- IV
CLUSTERING AND APPLICATIONS
Cluster analysis-Types of Data in Cluster Analysis-Categorization of Major Clustering Methods-Partitioning Methods,-Hierarchical Methods- Density-Based Methods,-Grid-Based Methods,-Model-

Based Clustering Methods- Clustering high dimensional data-Constraint- Based cluster analysis-Outlier Analysis

UNIT - V
MINING DATA STREAMS, TIME-SERIES AND SEQUENCE DATA
Basic concepts- Mining data streams-Mining Time-series data--Mining sequence patterns in Transactional databases-.Mining Object- Spatial- Multimedia-Text and Web data- Spatial Data mining- Multimedia Data mining--Text Mining- Mining the World Wide Web.

TEXT BOOKS:
1. Data Mining – Concepts and Techniques - JIAWEI HAN & MICHELINE KAMBER, Elsevier.
2. Data Warehousing, Data Mining &OLAP- Alex Berson and Stephen J. Smith- Tata McGraw-Hill Edition, Tenth reprint 2007

REFERENCES:
1. Building the Data Warehouse- W. H. Inmon, Wiley Dreamtech India Pvt. Ltd.
2. Data Mining Introductory and Advanced topics –MARGARET H DUNHAM, PEA.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS II Year II Semester

INFORMATION SECURITY (Core Elective – I)

Prerequisites
- A Course on "Computer Networks, Mathematics

Course Objectives
- Understand information security's importance in our increasingly computer-driven world..
- Master the key concepts of information security and how they "work."
- To understand the fundamentals of Cryptography
- To understand the various key distribution and management schemes
- To understand how to deploy encryption techniques to secure data in transit across data networks
- To apply algorithms used for secure transactions in real world applications

Course Outcomes
- Ability to demonstrate the knowledge of cryptography and network security concepts and applications.
- Apply security principles in system design.
- Ability to identify and investigate vulnerabilities and security threats and mechanisms to counter them.

UNIT - I
Security Attacks (Interruption, Interception, Modification and Fabrication), Security Services (Confidentiality, Authentication, Integrity, Non-repudiation, access Control and Availability) and Mechanisms, A model for Internetwork security.
Classical Encryption Techniques, DES, Strength of DES, Differential and Linear Cryptanalysis, Block Cipher Design Principles and Modes of operation, Blowfish, Placement of Encryption Function, Traffic Confidentiality, key Distribution, Random Number Generation.

UNIT - II
Public key Cryptography Principles, RSA algorithm, Key Management, Diffie-Hellman Key Exchange, Elliptic Curve Cryptography.
Message authentication and Hash Functions, Authentication Requirements and Functions, Message Authentication, Hash Functions and MACs Hash and MAC Algorithms SHA-512, HMAC.

UNIT - III
Digital Signatures, Authentication Protocols, Digital signature Standard, Authentication Applications, Kerberos, X.509 Directory Authentication Service.
Email Security: Pretty Good Privacy (PGP) and S/MIME.

UNIT - IV
IP Security:
Overview, IP Security Architecture, Authentication Header, Encapsulating Security Payload, Combining Security Associations and Key Management.
Web Security: Web Security Requirements, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET).

UNIT V

Intruders, Viruses and Worms Intruders, Viruses and related threats Firewalls: Firewall Design Principles, Trusted Systems, Intrusion Detection Systems.

TEXT BOOKS:
1. Cryptography and Network Security (principles and approaches) by William Stallings Pearson Education, 4th Edition.

REFERENCE BOOKS:
1. Network Security Essentials (Applications and Standards) by William Stallings Pearson Education.
2. Principles of Information Security, Whitman, Thomson.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS II Year II Semester

DISTRIBUTED SYSTEMS (CORE ELECTIVE - II)

Prerequisites:
- A course on " Operating Systems"

Course Objectives:
- This course provides an insight into Distributed systems.
- Topics include- Peer to Peer Systems, Transactions and Concurrency control, Security and Distributed shared memory

Course Outcomes:
- Ability to understand Transactions and Concurrency control.
- Ability to understand Security issues.
- Understanding Distributed shared memory.

UNIT - I
Characterization of Distributed Systems-Introduction, Examples of Distributed systems, Resource sharing and web, challenges, System models-Introduction, Architectural and Fundamental models, Networking and Internetworking, Interprocess Communication,
Distributed objects and Remote Invocation-Introduction, Communication between distributed objects, RPC, Events and notifications, Case study-Java RMI.

UNIT - II
Operating System Support- Introduction, OS layer, Protection, Processes and Threads, Communication and Invocation, Operating system architecture, Distributed File Systems-Introduction, File Service architecture, case study- SUN network file systems.
Name Services-Introduction; Name Services and the Domain Name System, Case study of the Global Name Service, Case study of the X.500 Directory Service.

UNIT - III
Peer to Peer Systems–Introduction, Napster and its legacy, Peer to Peer middleware, Routing overlays, Overlay case studies-Pastry, Tapestry, Application case studies-Squirrel, OceanStore.
Time and Global States-Introduction, Clocks, events and Process states, Synchronizing physical clocks, logical time and logical clocks, global states, distributed debugging.
Coordination and Agreement-Introduction, Distributed mutual exclusion, Elections, Multicast communication, consensus and related problems.

UNIT - IV
Transactions and Concurrency control-Introduction, Transactions, Nested Transactions, Locks, Optimistic concurrency control, Timestamp ordering, Comparison of methods for concurrency control. Distributed Transactions-Introduction, Flat and Nested Distributed Transactions, Atomic commit protocols, Concurrency control in distributed transactions, Distributed deadlocks, Transaction recovery, Replication-Introduction, System model and group communication, Fault tolerant services, Transactions with replicated data.

UNIT - V
Security-Introduction, Overview of Security techniques, Cryptographic algorithms, Digital signatures, Case studies-Kerberos, TLS, 802.11 WiFi.

Distributed shared memory, Design and Implementation issues, Sequential consistency and Ivy case study, Release consistency and Munin case study, Other consistency models, CORBA case study-Introduction, CORBA RMI, CORBA Services.

TEXT BOOKS:
1. Distributed Systems Concepts and Design, G Coulouris, J Dollimore and T Kindberg, Fourth Edition, Pearson Education.
2. Distributed Systems, S. Ghosh, Chapman & Hall/CRC, Taylor & Francis Group, 2010.

REFERENCE BOOKS:
1. Distributed Computing, S. Mahajan and S. Shah, Oxford University Press.
2. Distributed Operating Systems Concepts and Design, Pradeep K.Sinha, PHI.
3. Advanced Concepts in Operating Systems, M Singhal, N G Shivarathri, TMH.
4. Reliable Distributed Systems, K. P. Birman, Springer.
5. Distributed Systems – Principles and Paradigms, A.S. Tanenbaum and M.V. Steen, Pearson Education.
6. Distributed Operating Systems and Algorithm Analysis, R. Chow, T. Johnson, Pearson.
7. Distributed Operating Systems, A. S. Tanenbaum, Pearson education.
8. Distributed Computing, Principles, Algorithms and Systems, Ajay D. Kshemakalyani and Mukesh Singhal, Cambridge, rp 2010.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS II Year II Semester

SOFTWARE PROCESS AND PROJECT MANAGEMENT (CORE ELECTIVE – II)

Course Objectives: At the end of the course, the student shall be able to:
- To describe and determine the purpose and importance of project management from the perspectives of planning, tracking and completion of project.
- To compare and differentiate organization structures and project structures.
- To implement a project to manage project schedule, expenses and resources with the application of suitable project management tools.

UNIT - I
Software Process Maturity: Software maturity Framework, Principles of Software Process Change, Software Process Assessment, The Initial Process, The Repeatable Process, The Defined Process, The Managed Process, The Optimizing Process. **Process Reference Models Capability** Maturity Model (CMM), CMMi, PCMM, PSP, TSP.

UNIT - II
Software Project Management Renaissance Conventional Software Management, Evolution of Software Economics, Improving Software Economics, The old way and the new way. **Life-Cycle Phases and Process artifacts** Engineering and Production stages, inception phase, elaboration phase, construction phase, transition phase, artifact sets, management artifacts, engineering artifacts and pragmatic artifacts, model based software architectures.

UNIT - III
Workflows and Checkpoints of process Software process workflows, Iteration workflows, Major milestones, Minor milestones, Periodic status assessments. **Process Planning** Work breakdown structures, Planning guidelines, cost and schedule estimating process, iteration planning process, Pragmatic planning.

UNIT - IV
Project Organizations Line-of- business organizations, project organizations, evolution of organizations, process automation. **Project Control and process instrumentation** The seven core metrics, management indicators, quality indicators, life-cycle expectations, Pragmatic software metrics, metrics automation.

UNIT V
CCPDS-R Case Study and Future Software Project Management Practices Modern Project Profiles, Next-Generation software Economics, Modern Process Transitions.

TEXT BOOKS:
1. Managing the Software Process, *Watts S. Humphrey*, Pearson Education.
2. Software Project Management, *Walker Royce,* Pearson Education.

REFERENCE BOOKS:
1. Effective Project Management: Traditional, Agile, Extreme, Robert Wysocki, Sixth edition, Wiley India, rp2011.
2. An Introduction to the Team Software Process, Watts S. Humphrey, Pearson Education, 2000
3. Software Project Management, Bob Hughes & Mike Cotterell, fourth edition, TMH, 2006
4. Applied Software Project Management, Andrew Stellman & Jennifer Greene, O'Reilly, 2006.
5. Head First PMP, Jennifer Greene & Andrew Stellman, O'Reilly, 2007

6. Software Engineering Project Managent, Richard H. Thayer & Edward Yourdon, 2nd edition, Wiley India, 2004.
7. The Art of Project Management, Scott Berkun, SPD, O'Reilly, 2011.
8. Applied Software Project Management, Andrew Stellman & Jennifer Greene, SPD, O'Reilly, rp2011.
9. Agile Project Management, Jim Highsmith, Pearson education, 2004.
10. Process Improvement essentials, James R. Persse, O'Reilly, 2006

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS II Year II Semester

SOFT COMPUTING (CORE ELECTIVE – II)

Course Objectives
- This course explains AI Problems and Search techniques.
- To understand Supervised and Unsupervised Learning Networks
- Introduces Classical Sets and Fuzzy Sets.
- To understand Genetic Algorithms.

Course Outcomes
- Comprehend the differences between Classical Sets and Fuzzy Sets
- Get the skill for application of search techniques to solve AI Problems
- Able to apply Genetic Algorithms for solving real time problems

UNIT- I
AI Problems and Search: AI problems, Techniques, Problem Spaces and Search, Heuristic Search Techniques- Generate and Test, Hill Climbing, Best First Search Problem reduction, Constraint Satisfaction and Means End Analysis. Approaches to Knowledge Representation- Using Predicate Logic and Rules.

UNIT- II
Supervised Learning Networks-perceptron, Back propagation algorithm-Classification Problem-Speech Processing Case study.
Unsupervised Learning Network- Introduction, Fixed Weight Competitive Nets, Maxnet, Hamming Network, Kohonen Self-Organizing Feature Maps, Learning Vector Quantization,

UNIT - III
Introduction to Classical Sets (crisp Sets)and Fuzzy Sets- operations and Fuzzy sets. Classical Relations –and Fuzzy Relations- Cardinality, Operations, Properties and composition. Tolerance and equivalence relations.
Membership functions- Features, Fuzzification, membership value assignments, Defuzzification.

UNIT - IV
Fuzzy Arithmetic and Fuzzy Measures, Fuzzy Rule Base and Approximate Reasoning Fuzzy Decision making
Fuzzy Logic Control Systems. Genetic Algorithm- Introduction and basic operators and terminology.
Applications: Optimization of TSP, Internet Search Technique

UNIT - V
Genetic Algorithms-Introduction, Biological background, Search space, Basic technologies, Simple and general genetic algorithms, Operations in genetic algorithms, Stopping criteria and constraints in genetic algorithms, Problem solving using genetic algorithms.

TEXT BOOKS:
1. Principles of Soft Computing- S N Sivanandam, S N Deepa, Wiley India, 2007
2. Soft Computing and Intelligent System Design –Fakhreddine O Karray, Clarence D Silva,. Pearson Edition, 2004.

REFERENCE BOOKS:

1. Artificial Intelligence and Soft Computing- Behavioural and Cognitive Modelling of the Human Brain- Amit Konar, CRC press, Taylor and Francis Group.
2. Artificial Intelligence – Elaine Rich and Kevin Knight, TMH, 1991, rp2008.
3. Artificial Intelligence – Patric Henry Winston – Third Edition, Pearson Education.
4. A first course in Fuzzy Logic-Hung T Nguyen and Elbert A Walker, CRC. Press Taylor and Francis Group.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS II Year II Semester

WEB TECHNOLOGIES LAB

1. Develop static pages (using Only HTML) of an online Book store. The pages should resemble: www.amazon.com The website should consist the following pages.
2. Home page
3. Registration and user Login
4. User Profile Page
5. Books catalog
6. Shopping Cart
7. Payment By credit card
8. Order Conformation

2. Validate the Registration, user login, user profile and payment by credit card pages using JavaScript.

3. Create and save an XML document at the server, which contains 10 users information. Write a program, which takes User Id as an input and returns the user details by taking the user information from the XML document.

4. Bean Assignments

    a. Create a JavaBean which gives the exchange value of INR(Indian Rupees) into equivalent American/Canadian/Australian Dollar value.
    b. Create a simple Bean with a label – which is the *count* of number of clicks. Than create a BeanInfo class such that only the "*count*" property is visible in the Property Window.
    c. Create two Beans-a)KeyPad .b)DisplayPad .After that integrate the two Beans to make it work as a Calculator.
    d. Create two Beans Traffic Light(Implemented as a Label with only three background colours-Red,Green,Yellow) and Automobile(Implemented as a TextBox which states its state/movement). The state of the Automobile should depend on the following Light Transition Table.

Light Transition	Automobile State
Red -→ Yellow	Ready
Yellow -→ Green	Move
Green → Red	Stopped

5. Install TOMCAT web server. Convert the static web pages of assignments 2 into dynamic web pages using servlets and cookies. Hint: Users information (user id, password, credit card number) would be stored in web.xml. Each user should have a separate Shopping Cart.

6. Redo the previous task using JSP by converting the static web pages of assignments 2 into dynamic web pages. Create a database with user information and books information. The books catalogue should be dynamically loaded from the database. Follow the MVC architecture while doing the website.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS II Year II Semester

LINUX PROGRAMMING LAB

Linux Programming:
Note: Use Bash for Shell scripts.

1. Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.

2. Write a shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it.

3. Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.

4. Write a shell script that receives any number of file names as arguments checks if every argument supplied is a file or a directory and reports accordingly. Whenever the argument is a file, the number of lines on it is also reported.

5. Write a shell script that accepts a list of file names as its arguments, counts and reports the occurrence of each word that is present in the first argument file on other argument files.

6. Write a shell script to list all of the directory files in a directory.

7. Write a shell script to find factorial of a given integer.

8. Write an awk script to count the number of lines in a file that do not contain vowels.
9. Write an awk script to find the number of characters, words and lines in a file.
10. Write a C program that makes a copy of a file using standard I/O and system calls.
11. Implement in C the following Unix commands using System calls
    a). Cat   b) mv

12. Write a C program to list files in a directory.

13. Write a C program to emulate the Unix ls –l command.
14. Write a C program to list for every file in a directory, its inode number and file name.
15. Write a C program that redirects standard output to a file.Ex: ls > f1.
16. Write a C program to create a child process and allow the parent to display "parent" and the child to display "child" on the screen.
17. Write a C program to create a Zombie process.
18. Write a C program that illustrates how an orphan is created.
19. Write a C program that illustrates how to execute two commands concurrently with a command pipe. Ex :- ls –l | sort
20. Write C programs that illustrate communication between two unrelated processes using named pipe.
21. Write a C program in which a parent writes a message to a pipe and the child reads the message.
22. Write a C program (sender.c) to create a message queue with read and write permissions to write 3 messages to it with different priority numbers.

23. Write a C program (receiver.c) that receives the messages (from the above message queue as specified in (22)) and displays them.

24. Write a C programs to transfer a large amount of data between processes, using
    a) a pipe   b)a FIFO c)a message queue.

25. Write a C program to allow cooperating processes to lock a resource for exclusive use,
    using: a)Semaphores   b)flock or lockf system calls.

26. Write a C program that illustrates suspending and resuming processes using signals.

27. Write a C program that implements a producer-consumer system with two processes.
    (using Semaphores).

28. Write client and server programs (using c) for interaction between server and client processes using Unix Domain sockets.

29. Write client and server programs (using c) for interaction between server and client processes using Internet Domain sockets.

30. Write C programs that illustrate two processes communicating using shared memory.

TEXT BOOKS:
   1. Advanced Unix Programming, N. B. Venkateswarulu, BS Publications.
   2. Unix and Shell programming, B. A. Forouzan and R. F. Gilberg, Cengage Learning.
   3. Unix and Shell Programming, M.G. Venkatesh Murthy, Pearson Education, 2005.
   4. Unix Shells by Example, 4th Edition, Elllie Quigley, Pearson Education.
   5. Sed and Awk, O. Dougherty & A. Robbins, 2nd edition, SPD.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS III Year I Semester

MOBILE APPLICATION DEVELOPMENT

Prerequisites:
- A Course on JAVA
- A Course on DBMS

Course Objectives:
- To demonstrate their understanding of the fundamentals of Android operating systems
- To demonstrate their skills of using Android software development tools
- To demonstrate their ability to develop software with reasonable complexity on mobile platform
- To demonstrate their ability to deploy software to mobile devices
- To demonstrate their ability to debug programs running on mobile devices

Course Outcomes:
- Student understands the working of Android OS Practically.
- Student will be able to develop, deploy and maintain the Android Applications.

UNIT - I:
Introduction to Android Operating System: Android OS design and Features – Android development framework, SDK features, Installing and running applications on Eclipse platform, Creating AVDs, Types of Android applications, Best practices in Android programming, Android tools
Android application components – Android Manifest file, Externalizing resources like values, themes, layouts, Menus etc, Resources for different devices and languages, Runtime Configuration Changes
Android Application Lifecycle – Activities, Activity lifecycle, activity states, monitoring state changes

UNIT - II:
Android User Interface: Measurements – Device and pixel density independent measuring units
Layouts – Linear, Relative, Grid and Table Layouts
User Interface (UI) Components – Editable and non editable TextViews, Buttons, Radio and Toggle Buttons, Checkboxes, Spinners, Dialog and pickers
Event Handling – Handling clicks or changes of various UI components
Fragments – Creating fragments, Lifecycle of fragments, Fragment states, Adding fragments to Activity, adding, removing and replacing fragments with fragment transactions, interfacing between fragments and Activities, Multi-screen Activities

UNIT - III
Intents and Broadcasts: Intent – Using intents to launch Activities, Explicitly starting new Activity, Implicit Intents, Passing data to Intents, Getting results from Activities, Native Actions, using Intent to dial a number or to send SMS
Broadcast Receivers – Using Intent filters to service implicit Intents, Resolving Intent filters, finding and using Intents received within an Activity
Notifications – Creating and Displaying notifications, Displaying Toasts

UNIT - IV
Persistent Storage: Files – Using application specific folders and files, creating files, reading data from files, listing contents of a directory Shared Preferences – Creating shared preferences, saving and retrieving data using Shared Preference

Database – Introduction to SQLite database, creating and opening a database, creating tables, inserting retrieving and deleting data, Registering Content Providers, Using content Providers (insert, delete, retrieve and update)

UNIT - V
Advanced Topics: Alarms – Creating and using alarms
Using Internet Resources – Connecting to internet resource, using download manager
Location Based Services – Finding Current Location and showing location on the Map, updating location

TEXT BOOKS:
1. Professional Android 4 Application Development, Reto Meier, Wiley India, (Wrox) , 2012
2. Android Application Development for Java Programmers, James C Sheusi, Cengage Learning, 2013

REFERENCES:
1. Beginning Android 4 Application Development, Wei-Meng Lee, Wiley India (Wrox), 2013

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS III Year I Semester

SOFTWARE TESTING METHODOLOGIES

Prerequisites:
- A course on "Software Engineering"

Course Objectives:
- To provide knowledge of the concepts in software testing such as testing process, criteria, strategies, and methodologies.
- To develop skills in software test automation and management using latest tools.

Course Outcomes:
- Design and develop the best test strategies in accordance to the development model.

UNIT - I:
Introduction: - Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs
Flow graphs and Path testing:- Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

UNIT - II:
Transaction Flow Testing:-transaction flows, transaction flow testing techniques.
Dataflow testing:- Basics of data flow testing, strategies in data flow testing, application of dataflow testing.
Domain Testing:-domains and paths, nice & ugly domains, domain testing, domains and interfaces' testing, domain and interface testing, domains and testability.

UNIT - III:
Paths, Path products and Regular expressions:- path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.
Logic Based Testing:- overview, decision tables, path expressions, kv charts, specifications.

UNIT - IV:
State, State Graphs and Transition testing:- state graphs, good & bad state graphs, state testing, Testability tips.

UNIT - V:
Graph Matrices and Application:-Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools. (Student should be given an exposure to a tool like Jmeter or Win-runner).

TEXT BOOKS:
1. Software Testing techniques – Baris Beizer, Dreamtech, second edition.
2. Software Testing Tools – Dr. K. V. K. K. Prasad, Dreamtech.

REFERENCES:
1. The craft of software testing – Brian Marick, Pearson Education.
3. Software Testing Techniques – SPD(Oreille)
4. Software Testing in the Real World – Edward Kit, Pearson.
5. Effective methods of Software Testing, Perry, John Wiley.
6. Art of Software Testing – Meyers, John Wiley.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS III Year I Semester

WEB SERVICES & SERVICE ORIENTED ARCHITECTURE (CORE Elective - III)

Pre-requisites:
- The course assumes a reasonable comfort and background about Information Technology and Management Information Systems.

Course Objectives:
- To gain understanding of the basic principles of service orientation
- To learn service oriented analysis techniques
- To learn technology underlying the service design
- To learn the concepts such as SOAP, Registering and Discovering Services.

Course Outcomes: At the end of this course, students are expected to gain the following learning:
- Get the foundations and concepts of service based computing
- Advocate the importance and means of technology alignment with business
- Understanding the basic operational model of web services,
- Gain the knowledge of key technologies in the service oriented computing arena
- Apply and practice the learning through a real or illustrative project/case study.

UNIT - I
Evolution and Emergence of Web Services – Evolution of distributed computing. Core distributed computing technologies – client/server, CORBA, JAVA RMI, Micro Soft DCOM, MOM, Challenges in Distributed Computing, role of J2EE and XML in distributed computing, emergence of Web Services and Service Oriented Architecture (SOA). Introduction to Web Services – The definition of web services, basic operational model of web services, tools and technologies enabling web services, benefits and challenges of using web services.

UNIT - II
Web Service Architecture – Web services Architecture and its characteristics, core building blocks of web services, standards and technologies available for implementing web services, web services communication, basic steps of implementing web services. Describing Web Services – WSDL introduction, non functional service description, WSDL1.1 Vs WSDL 2.0, WSDL document, WSDL elements, WSDL binding, WSDL tools, WSDL port type, limitations of WSDL.

UNIT - III
Brief Over View of XML – XML Document structure, XML namespaces, Defining structure in XML documents, Reuse of XML schemes, Document navigation and transformation. SOAP : Simple Object Access Protocol, Inter-application communication and wire protocols, SOAP as a messaging protocol, Structure of a SOAP message, SOAP envelope, Encoding, Service Oriented Architectures, SOA revisited, Service roles in a SOA, Reliable messaging, The enterprise Service Bus, SOA Development Lifecycle, SOAP HTTP binding, SOAP communication model, Error handling in SOAP.

UNIT – IV
Registering and Discovering Services : The role of service registries, Service discovery, Universal Description, Discovery, and Integration, UDDI Architecture, UDDI Data Model, Interfaces, UDDI Implementation, UDDI with WSDL, UDDI specification, Service Addressing and Notification, Referencing and addressing Web Services, Web Services Notification.

UNIT - V

SOA and web services security considerations, Network-level security mechanisms, Application-level security topologies, XML security standards, Semantics and Web Services, The semantic interoperability problem, The role of metadata, Service metadata, Overview of .NET and J2EE, SOA and Web Service Management, Managing Distributed System, Enterprise management Framework, Standard distributed management frameworks, Web service management, Richer schema languages, WS-Metadata Exchange.

TEXT BOOKS:
1. Web Services & SOA Principles and Technology, Second Edition, Michael P. Papazoglou.
2. Developing Java Web Services, R. Nagappan, R. Skoczylas, R.P. Sriganesh, Wiley India.
3. Developing Enterprise Web Services, S. Chatterjee, J. Webber, Pearson Education.

REFERENCE BOOKS:
1. XML, Web Services, and the Data Revolution, F. P. Coyle, Pearson Education.
2. Building web Services with Java, 2nd Edition, S. Graham and others, Pearson Education.
3. Java Web Services, D.A. Chappell & T. Jewell, O'Reilly, SPD.
4. McGovern, et al., "Java web Services Architecture", Morgan Kaufmann Publishers, 2005.
5. J2EE Wer Services, Richard Monson-Haefel, Pearson Education.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS III Year I Semester

DISTRIBUTED DATABASES (CORE ELECTIVE - III)

Prerequisites
- A course on "Database Management Systems"

Course Objectives
- To acquire knowledge on parallel and distributed databases and its applications.
- To study the usage and applications of Object Oriented databases.
- To learn the modeling and design of databases
- To acquire knowledge on parallel and distributed databases and its applications.
- Equip students with principles and knowledge of parallel and object oriented databases.
- Topics include distributed DBMS architecture and design; query processing and optimization; distributed transaction management and reliability; parallel and object database management systems.

Course Outcomes
- Understand theoretical and practical aspects of distributed database systems.
- Study and identify various issues related to the development of distributed database system.
- Understand the design aspects of object oriented database system and related development.

UNIT - I
Features of Distributed versus Centralized Databases, Principles of Distributed Databases, Levels Of Distribution Transparency, Reference Architecture for Distributed Databases, Types of Data Fragmentation, Integrity Constraints in Distributed Databases, Distributed Database Design

UNIT - II
Translation of Global Queries to Fragment Queries,   Equivalence transformations for Queries, Transforming Global Queries into Fragment Queries, Distributed Grouping and Aggregate Function Evaluation, Parametric Queries.
Optimization of Access Strategies, A Framework for Query Optimization, Join Queries, General Queries

UNIT - III
The Management of Distributed Transactions, A Framework for Transaction Management, Supporting Atomicity of Distributed Transactions, Concurrency Control for Distributed Transactions, Architectural Aspects of Distributed Transactions
Concurrency Control,   Foundation of Distributed Concurrency Control, Distributed Deadlocks, Concurrency Control based on Timestamps, Optimistic Methods for Distributed Concurrency Control.

UNIT - IV
Reliability, Basic Concepts, Nonblocking Commitment Protocols, Reliability and concurrency Control, Determining a Consistent View of the Network, Detection and Resolution of Inconsistency, Checkpoints and Cold Restart, Distributed Database Administration, Catalog Management in Distributed Databases, Authorization and Protection

UNIT - V
Architectural Issues, Alternative Client/Server Architectures, Cache Consistency, Object Management, Object Identifier Management, Pointer Swizzling, Object Migration, Distributed Object Storage, Object Query Processing, Object Query Processor Architectures, Query Processing Issues,

Query Execution, Transaction Management, Transaction Management in Object DBMSs, Transactions as Objects

Database Integration, Scheme Translation, Scheme Integration, Query Processing Query Processing Layers in Distributed Multi-DBMSs, Query Optimization Issues Transaction Management Transaction and Computation Model, Multidatabase Concurrency Control, Multidatabase Recovery, Object Orientation and Interoperability, Object Management Architecture CORBA and Database interoperability, Distributed Component Object Model, COM/OLE and Database Interoperability, PUSH-Based Technologies

TEXT BOOKS:
1. Distributed Databases Principles & Systems, Stefano Ceri, Giuseppe Pelagatti, TMH.
2. Principles of Distributed Database Systems, M. Tamer Ozsu, Patrick Valduriez , Pearson Education, 2nd Edition.

REFERENCE BOOKS:
1. Distributed Database Systems, Chanda Ray, Pearson.
2. Distributed Database Management Systems, S. K. Rahimi and Frank. S. Haug, Wiley.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS III Year I Semester

INFORMATION RETRIEVAL SYSTEMS (CORE ELECTIVE - III)

Prerequisites:
- Data Structures

Course Objectives:
- to learn the important concepts and algorithms
- To understand the data/file structures those are necessary to design, and implement Information Retrieval (IR) systems.

Course Outcomes:
- Can be able to use different information retrieval techniques in various application areas
- Can apply IR principles to locate relevant information large collections of data
- Able to analyse performance of retrieval systems when dealing with unmanaged data sources
- Can be able to implement retrieval systems for web search tasks.

UNIT - I
Introduction:
Motivation, Basic Concepts, Past-Present and Future, the Retrieval Process
Modeling :
Introduction, A Taxonomy of Information retrieval Models, Retrieval: Ad hoc and Filtering, A Formal Characteristics of IR Models, Classic Information Retrieval, Alternative Ste Theory Models, Alternative Probabilistic Models, Structured Text Retrieval Models, Model for Browsing

UNIT - II
Retrieval Evaluation
Introduction, retrieval Performance Evaluation, Reference Collections
Query languages
Introduction, Keyword-Based Querying, Pattern Matching, Structural Queries, Query Protocols
Query Operations
Introduction, User Relevance Feedback, Automatic Local Analysis, Automatic global Analysis
Text Operations
Introduction, Document Preprocessing, Document Clustering, Text Compression, Comparing text Compression Techniques

UNIT - III
Indexing and Searching
Introduction, Inverted Files, Other Indices for Text, Boolean queries, Sequential Searching, pattern Matching, Structural Queries, Compression
Searching the Web
Introduction, Challenges, Characterizing the Web, Search Engines, Browsing, Metasearches, Finding the Needle in the Haystack, Searching using Hyperlinks

UNIT - IV
User Interfaces and Visualization
Introduction, human-Computer Interaction, The Information Access Process, Starting Points, Query Specification, Context, User Relevance Judgments, Interface Support for the Search Process

UNIT - V

Multimedia IR: Models and Languages

Introduction, Data Modeling, Query Languages

Multimedia IR: Indexing and |Searching

Introduction, Background-Spatial Access Methods,A Generic Multimedia Indexing Approach, One Dimentional Time Series, wo dimential Color Images, Automatic Feature Extraction.

TEXT BOOKS:
1. Modern Information Retrival by Yates and Neto Pearson Education.

REFERENCES:
1. Kowalski, Gerald, Mark T Maybury: Information Retrieval Systems: Theory and Implementation, Kluwer Academic Press, 1997.
2. Frakes, W.B., Ricardo Baeza-Yates: Information Retrieval Data Structures and Algorithms, Prentice Hall, 1992.
3. Information Storage & Retieval By Robert Korfhage – John Wiley & Sons.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS III Year I Semester

ETHICAL HACKING (CORE ELECTIVE - IV)

Prerequisites
- A course on "Operating Systems"
- A course on "Computer Networks"
- A course on "Network Security and Cryptography"

Course Objectives
- The aim of the course is to introduce the methodologies and framework of ethical hacking for enhancing the security.
- The course includes-Impacts of Hacking; Types of Hackers; Information Security Models; Information Security Program; Business Perspective; Planning a Controlled Attack; Framework of Steps (Reconnaissance, Enumeration, Vulnerability Analysis, Exploitation, Deliverable and Integration)

Course Outcomes
- Gain the knowledge of the use and availability of tools to support an ethical hack
- Gain the knowledge of interpreting the results of a controlled attack
- Understand the role of politics, inherent and imposed limitations and metrics for planning of a test
- Comprehend the dangers associated with penetration testing

UNIT - I
Introduction: Hacking Impacts, The Hacker
Framework: Planning the test, Sound Operations, Reconnaissance, Enumeration, Vulnerability Analysis, Exploitation, Final Analysis, Deliverable, Integration
Information Security Models: Computer Security, Network Security, Service Security, Application Security, Security Architecture
Information Security Program: The Process of Information Security, Component Parts of Information Security Program, Risk Analysis and Ethical Hacking

UNIT - II
The Business Perspective: Business Objectives, Security Policy, Previous Test Results, Business Challenges
Planning for a Controlled Attack: Inherent Limitations, Imposed Limitations, Timing is Everything, Attack Type, Source Point, Required Knowledge, Multi-Phased Attacks, Teaming and Attack Structure, Engagement Planner, The Right Security Consultant, The Tester, Logistics, Intermediates, Law Enforcement

UNIT - III
Preparing for a Hack: Technical Preparation, Managing the Engagement
Reconnaissance: Social Engineering, Physical Security, Internet Reconnaissance

UNIT - IV
Enumeration: Enumeration Techniques, Soft Objective, Looking Around or Attack, Elements of Enumeration, Preparing for the Next Phase
Exploitation: Intuive Testing, Evasion, Threads and Groups, Operating Systems, Password Crackers, RootKits, applications, Wardialing, Network, Services and Areas of Concern

UNIT - V
Deliverable: The Deliverable, The Document, Overal Structure, Aligning Findings, Presentation
Integration: Integrating the Results, Integration Summary, Mitigation, Defense Planning, Incident Management, Security Policy, Conclusion

TEXT BOOK:
1. James S. Tiller, "The Ethical Hack: A Framework for Business Value Penetration Testing", Auerbach Publications, CRC Press

REFERENCE BOOKS:
1. EC-Council, "Ethical Hacking and Countermeasures Attack Phases", Cengage Learning
2. Michael Simpson, Kent Backman, James Corley, "Hands-On Ethical Hacking and Network Defense", Cengage Learning

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS III Year I Semester

MULTIMEDIA AND RICH INTERNET APPLICATIONS (CORE ELECTIVE – IV)

Course Objectives:
This course aims to further develop students' competency in producing dynamic and creative graphic solutions for multimedia productions. It provides students with the basic concepts and techniques of interactive authoring. It also introduces students with the advanced scripting skills necessary for implementing highly interactive, rich internet applications using multimedia technologies and authoring tools. Students will develop aesthetic value and competencies in multimedia authoring. Artistic visual style and layout design are stressed, as well as the editing and integration of graphic images, animation, video and audio files. The course allows students to master industry-wide software and technologies to create highly interactive, rich internet applications.

UNIT - I
Introduction to Multimedia: Internet and Multimedia communications, Multimedia Networks, Multimedia Applications, Multimedia Information representation- Digitization Principles, Text, Images, Audio and Video, Compression Methods-Basic Coding Methods – Run Length coding, Huffman coding, Arithmetic coding, Discrete Cosine Transform, Differential PCM, Motion Compensated Prediction, Video Compression – JPEG, H.261, MPEG-1 Video, MPEG 2 and 3 Video, H.263, Wavelet and Fractal Image Compression, Audio Compression.

UNIT - II
Multimedia Applications in Networks: Introduction, Application Level Framing, Audio/Video Conferencing-Session Directories, Audio/Video Conferencing, Adaptive Applications, Receiver Heterogeneity, Real Time Application with Resource Reservation ,Video Server, Applications requiring reliable multicast – White Board , Network Text Editor for Shared Text Editing, Multi Talk, Multicast file transfer, Multimedia Applications on the World Wide Web – Multicast Web Page Sharing, Audio/Video Streams in the www, Interactive Multiplayer Games.

UNIT - III
Web 2.0: What is web 2.0, Search, Content Networks, User Generated Content, Blogging, Social Networking, Social Media, Tagging, Social Marking, Rich Internet Applications, Web Services, Mashups, Location Based Services, XML, RSS, Atom, JSON, and VoIP, Web 2.0 Monetization and Business Models, Future of the Web.

UNIT - IV
Rich Internet Applications (RIAs) with Adobe Flash and Flex: Adobe Flash- Introduction, Flash Movie Development, Learning Flash with Hands-on Examples, Publish your flash movie, Creating special effects with Flash, Creating a website splash screen, action script, web sources.
Adobe Flex 2- Introduction, Flex Platform Overview, Creating a Simple User Interface, Accessing XML data from your application, Interacting with Server Side Applications, Customizing your User Interface, Creating Charts and Graphs, Connection Independent RIAs on the desktop -Adobe Integrated Runtime (AIR), Flex 3 Beta.

UNIT - V
Ajax- Enabled Rich Internet Application: Introduction, Traditional Web Applications Vs Ajax Applications, Rich Internet Application with Ajax, History of Ajax, Raw Ajax example using xml http request object, Using XML, Creating a full scale Ajax Enabled application, Dojo ToolKit.

TEXT BOOKS:
1. Multimedia Communications: Protocols and Applications, Franklin F Kuo, J. Joaquin Garcia, Wolf gang Effelsberg, Prentice Hall Publications.
2. Multimedia Communications: Applications, Networks, Protocols and Standards, Fred Halsall, Addison Wesley Publications.
3. AJAX, Rich Internet Applications, and Web Development for Programmers, Paul J Deitel and Harvey M Deitel, Deitel Developer Series, Pearson education.

REFERENCE BOOKS:
1. Professional Adobe Flex 2, Rich Tretola, Simon barber and Renaun Erickson, Wrox, Wiley India Edition.
2. Multimedia Information Networking, Nalin K Sharda, PHI Learning.
3. Multimedia Computing, Communications & Applications, Ralf Steinmetz and Klara Nahrstedt, Pearson Education.
4. Multimedia Communication Systems: techniques, standards and networks, K.R.Rao, Bojkovic and Milovanovic, PHI Learning.
5. Programming Flex 3, C. Kazoun and J. Lott, SPD.
6. Dojo, J. E. Harmon, Pearson Education.
7. Adobe Flex 3: Training from the Source, Tapper & others, Pearson Education.
8. Principles of Multimedia, R. Parekh, TMH.
9. Mastering Dojo, R. Gill, C. Riecke and A. Russell, SPD.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS III Year I Semester

SEMANTIC WEB AND SOCIAL NETWORKS (CORE ELECTIVE - IV)

Course Objectives:
- To learn Web Intelligence
- To learn Knowledge Representation for the Semantic Web
- To learn Ontology Engineering
- To learn Semantic Web Applications, Services and Technology
- To learn Social Network Analysis and semantic web

UNIT – I:
Web Intelligence: Thinking and Intelligent Web Applications, The Information Age ,The World Wide Web, Limitations of Today's Web, The Next Generation Web, Machine Intelligence, Artificial Intelligence, Ontology, Inference engines, Software Agents, Berners-Lee www, Semantic Road Map, Logic on the semantic Web.

UNIT - II:
Knowledge Representation for the Semantic Web: Ontologies and their role in the semantic web, Ontologies Languages for the Semantic Web –Resource Description Framework(RDF) / RDF Schema, Ontology Web Language(OWL), UML, XML/XML Schema.

UNIT - III:
Ontology Engineering: Ontology Engineering, Constructing Ontology, Ontology Development Tools, Ontology Methods, Ontology Sharing and Merging, Ontology Libraries and Ontology Mapping, Logic, Rule and Inference Engines.

UNIT - IV:
Semantic Web Applications, Services and Technology: Semantic Web applications and services, Semantic Search, e-learning, Semantic Bioinformatics, Knowledge Base ,XML Based Web Services, Creating an OWL-S Ontology for Web Services, Semantic Search Technology, Web Search Agents and Semantic Methods,

UNIT - V:
Social Network Analysis and semantic web: What is social Networks analysis, development of the social networks analysis, Electronic Sources for Network Analysis – Electronic Discussion networks, Blogs and Online Communities, Web Based Networks. Building Semantic Web Applications with social network features.

TEXT BOOKS:
1. Thinking on the Web - Berners Lee, Godel and Turing, Wiley inter science, 2008.
2. Social Networks and the Semantic Web, Peter Mika, Springer, 2007.

REFERENCE BOOKS:
1. Semantic Web Technologies, Trends and Research in Ontology Based Systems, J. Davies, R. Studer, P.Warren, John Wiley & Sons.
2. Semantic Web and Semantic Web Services -Liyang Lu Chapman and Hall/CRC Publishers,(Taylor & Francis Group)
3. Information sharing on the semantic Web - Heiner Stuckenschmidt; Frank Van Harmelen, Springer Publications.
4. Programming the Semantic Web, T. Segaran, C.Evans, J.Taylor, O'Reilly, SPD.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS III Year I Semester

MOBILE APPLICATION DEVELOPMENT LAB

Course Objectives:
- To learn how to develop Applications in android environment.
- To learn how to develop user interface applications.
- To learn how to develop URL related applications.

Course Outcomes:
- Student understands the working of Android OS Practically.
- Student will be able to develop, deploy and maintain the Android Applications.

The student is expected to be able to do the following problems, though not limited.

1. Create an Android application that shows Hello + name of the user and run it on an emulator. (b) Create an application that takes the name from a text box and shows hello message along with the name entered in text box, when the user clicks the OK button.

2. Create a screen that has input boxes for User Name, Password, Address, Gender (radio buttons for male and female), Age (numeric), Date of Birth (Date Picket), State (Spinner) and a Submit button. On clicking the submit button, print all the data below the Submit Button. Use (a) Linear Layout (b) Relative Layout and (c) Grid Layout or Table Layout.

3. Develop an application that shows names as a list and on selecting a name it should show the details of the candidate on the next screen with a "Back" button. If the screen is rotated to landscape mode (width greater than height), then the screen should show list on left fragment and details on right fragment instead of second screen with back button. Use Fragment transactions and Rotation event listener.

4. Develop an application that uses a menu with 3 options for dialing a number, opening a website and to send an SMS. On selecting an option, the appropriate action should be invoked using intents.

5. Develop an application that inserts some notifications into Notification area and whenever a notification is inserted, it should show a toast with details of the notification.

6. Create an application that uses a text file to store user names and passwords (tab separated fields and one record per line). When the user submits a login name and password through a screen, the details should be verified with the text file data and if they match, show a dialog saying that login is successful. Otherwise, show the dialog with Login Failed message.

7. Create a user registration application that stores the user details in a database table.

8. Create a database and a user table where the details of login names and passwords are stored. Insert some names and passwords initially. Now the login details entered by the user should be verified with the database and an appropriate dialog should be shown to the user.

9. Create an admin application for the user table, which shows all records as a list and the admin can select any record for edit or modify. The results should be reflected in the table.

10. Develop an application that shows all contacts of the phone along with details like name, phone number, mobile number etc.

11. Create an application that saves user information like name, age, gender etc. in shared preference and retrieves them when the program restarts.

12. Create an alarm that rings every Sunday at 8:00 AM. Modify it to use a time picker to set alarm time.

13. Create an application that shows the given URL (from a text field) in a browser.

    Develop an application that shows the current location's latitude and longitude continuously as the device is moving (tracking).

14. Create an application that shows the current location on Google maps.

Note:

Android Application Development with MIT App Inventor:For the first one week, the student is advised to go through the App Inventor from MIT which gives insight into the various properties of each component.

The student should pay attention to the properties of each components, which are used later in Android programming. Following are useful links:

1. http://ai2.appinventor.mit.edu
2. https://drive.google.com/file/d/0B8rTtW_91YclTWF4czdBMEpZcWs/view

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS III Year I Semester

SOFTWARE ENGINEERING METHODOLOGIES AND UML LAB

PART-A
USING RATIONAL ROSE - CASE TOOL

LIST OF EXPERIMENTS

1. Prepare the following documents for two or three of the experiments listed below and develop the software engineering methodology.
2. Program Analysis and Project Planning. Thorough study of the problem - Identify project scope, Objectives, Infrastructure.
3. Software requirement Analysis. Describe the individual Phases / Modules of the project, Identify deliverables.
4. Data Modeling. Use work products - Data dictionary, Use diagrams and activity diagrams, build and test lass diagrams, Sequence diagrams and add interface to class diagrams.
5. Software Development and Debugging
6. Software Testing. Prepare test plan, perform validation testing, Coverage analysis, memory leaks, develop test case hierarchy, Site check and Site monitor.
7. Using WinRunner Design a test plan for Library Management System.

SUGGESTED LIST OF APPLICATIONS
1. Student Marks Analyzing System
2. Quiz System
3. Online Ticket Reservation System
4. Payroll System
5. Course Registration System
6. Expert Systems
7. ATM Systems
8. Stock Maintenance
9. Real-Time Scheduler
10. Remote Procedure Call Implementation
11. Library Management System

PART-B
UML LAB

Course Objectives:
The student should take up the case study of Unified Library application which is mentioned in the theory, and Model it in different views i.e Use case view, logical view, component view, Deployment view, Database design, forward and Reverse Engineering, and Generation of documentation of the project.

Student has to take up another case study of his/her own interest and do the same whatever mentioned in first problem. Some of the ideas regarding case studies are given in reference books which were mentioned in theory syllabus can be referred for some idea.

Problems on UML:

Unified Modeling Language Lab:
Students are divided into batches of 5 each and each batch has to draw the following diagrams using UML for an ATM system whose description is given below.

UML diagrams to be developed are:
1. Use Case Diagram.
2. Class Diagram.
3. Sequence Diagram.
4. Collaboration Diagram.
5. State Diagram
6. Activity Diagram.
7. Component Diagram
8. Deployment Diagram.
9. Test Design.

Description for an ATM System:
          The software to be designed will control a simulated automated teller machine (ATM) having a magnetic stripe reader for reading an ATM card, a customer console (keyboard and display) for interaction with the customer, a slot for depositing envelopes, a dispenser for cash (in multiples of Rs. 100, Rs. 500 and Rs. 1000), a printer for printing customer receipts, and a key-operated switch to allow an operator to start or stop the machine. The ATM will communicate with the bank's computer over an appropriate communication link. (The software on the latter is not part of the requirements for this problem.)

The ATM will service one customer at a time. A customer will be required to insert an ATM card and enter a personal identification number (PIN) - both of which will be sent to the bank for validation as part of each transaction. The customer will then be able to perform one or more transactions. The card will be retained in the machine until the customer indicates that he/she desires no further transactions, at which point it will be returned - except as noted below.

The ATM must be able to provide the following services to the customer:
1. A customer must be able to make a cash withdrawal from any suitable account linked to the card, in multiples of Rs. 100 or Rs. 500 or Rs. 1000. Approval must be obtained from the bank before cash is dispensed.
2. A customer must be able to make a deposit to any account linked to the card, consisting of cash and/or checks in an envelope. The customer will enter the amount of the deposit into the ATM, subject to manual verification when the envelope is removed from the machine by an operator. Approval must be obtained from the bank before physically accepting the envelope.
3. A customer must be able to make a transfer of money between any two accounts linked to the card.
4. A customer must be able to make a balance inquiry of any account linked to the card.
5. A customer must be able to abort a transaction in progress by pressing the Cancel key instead of responding to a request from the machine.

The ATM will communicate each transaction to the bank and obtain verification that it was allowed by the bank. Ordinarily, a transaction will be considered complete by the bank once it has been approved. In the case of a deposit, a second message will be sent to the bank indicating that the customer has deposited the envelope. (If the customer fails to deposit the envelope within the timeout period, or presses cancel instead, no second message will be sent to the bank and the deposit will not be credited to the customer.)

If the bank determines that the customer's PIN is invalid, the customer will be required to re-enter the PIN before a transaction can proceed. If the customer is unable to successfully enter the PIN after three tries, the card will be permanently retained by the machine, and the customer will have to contact the bank to get it back.

If a transaction fails for any reason other than an invalid PIN, the ATM will display an explanation of the problem, and will then ask the customer whether he/she wants to do another transaction. The ATM will provide the customer with a printed receipt for each successful transaction

The ATM will have a key-operated switch that will allow an operator to start and stop the servicing of customers. After turning the switch to the "on" position, the operator will be required to verify and enter the total cash on hand. The machine can only be turned off when it is not servicing a customer. When the switch is moved to the "off" position, the machine will shut down, so that the operator may remove deposit envelopes and reload the machine with cash, blank receipts, etc.

REFERENCE BOOKS:
1. Learning UML 2.0, Russ Miles and Kim Hamilton, O'Reilly, SPD.
2. Mastering UML with Rational Rose, W. Boggs & M. Boggs, Wiley India.