COURSE STRUCTURE

Master of Computer Applications

..I.. Year ..I..Semester

S. No.	Subject Code	Subject Title	Periods per week			C	Scheme of Examination Maximum Marks		
			T	P	D		Int.	Ext.	Total
1.	MCA101	C Programming and Data Structures	5	-	-	4	40	60	100
2.	MCA102	Digital Logic and Computer Organization	5	-	-	4	40	60	100
3.	MCA103	Discrete Mathematical Structures and Graph Theory	5	-	-	4	40	60	100
4.	MCA104	Probability and Statistical Applications	5	-	-	4	40	60	100
5.	MCA105	Accounting and Financial Management	5	-	-	4	40	60	100
6.	MCA106	English Language Communication Skills Lab	-	3	-	2	50	50	100
7.	MCA107	C Programming and Data Structures Lab	-	3	-	2	50	50	100
8.	MCA108	Digital Logic and Computer Organization Lab		3	-	2	50	50	100
		Total	**25**	**9**		**26**	**350**	**450**	**800**

T- THEORY P – PRACTICAL D- DRAWING C – CREDITS Int. – INTERNAL Ext. - EXTERNAL

COURSE STRUCTURE

Master of Computer Applications

..I.. Year ..II..Semester

S. No.	Subject Code	Subject Title	Periods per week			C	Scheme of Examination Maximum Marks		
			T	P	D		Int.	Ext.	Total
1.	MCA201	Object Oriented Programming with C++	5	-	-	4	40	60	100
2.	MCA202	UNIX & Network Programming	5	-	-	4	40	60	100
3.	MCA203	Operating Systems	5	-	-	4	40	60	100
4.	MCA204	Computer Networks	5	-	-	4	40	60	100
5.	MCA205	File Structure	5	-	-	4	40	60	100
6.	MCA206	Object Oriented Programming using C++ Lab	-	3	-	2	50	50	100
7.	MCA207	UNIX & Network Programming Lab	-	3	-	2	50	50	100
		Total	**25**	**6**		**24**	**300**	**400**	**700**

T- THEORY P – PRACTICAL D- DRAWING C – CREDITS Int. – INTERNAL Ext. - EXTERNAL

COURSE STRUCTURE

Master of Computer Applications

..II.. Year ..III..Semester

S. No.	Subject Code	Subject Title	Periods per week			C	Scheme of Examination Maximum Marks		
			T	P	D		Int.	Ext.	Total
1.	MCA301	Java Programming	5	-	-	4	40	60	100
2.	MCA302	Data Base Management Systems	5	-	-	4	40	60	100
3.	MCA303	Intellectual Property Rights & Professional Ethics	5	-	-	4	40	60	100
4.	MCA304	Computer Graphics & Vision	5	-	-	4	40	60	100
5.	MCA305	Design & Analysis of Algorithms	5	-	-	4	40	60	100
6.	MCA306	Java Programming Lab	-	3	-	2	50	50	100
7.	MCA307	Data Base Management Systems Lab	-	3	-	2	50	50	100
		Total	**25**	**6**		**24**	**300**	**400**	**700**

T- THEORY P – PRACTICAL D- DRAWING C – CREDITS Int. – INTERNAL Ext. - EXTERNAL

COURSE STRUCTURE

Master of Computer Applications

..II.. Year ..IV..Semester

S. No.	Subject Code	Subject Title	Periods per week			C	Scheme of Examination Maximum Marks		
			T	P	D		Int.	Ext.	Total
1.	MCA401	Web Technologies	5	-	-	4	40	60	100
2.	MCA402	Software Engineering	5	-	-	4	40	60	100
3.	MCA403	Data Warehousing & Data Mining	5	-	-	4	40	60	100
4.		Elective - 1	5	-	-	4	40	60	100
5.		Elective - 2	5	-	-	4	40	60	100
6.	MCA410	Data Warehousing & Data Mining Lab	-	3	-	2	50	50	100
7.	MCA411	Web Technologies Lab	-	3	-	2	50	50	100
8.	MCA412	Mini Project		2		2	50		50
		~~MCA333~~ TOTAL	**25**	**8**		**26**	**350**	**400**	**750**

T- THEORY P – PRACTICAL D- DRAWING C – CREDITS Int. – INTERNAL Ext. - EXTERNAL

LIST OF ELECTIVES

Department Elective 1:

a) Human Computer Interaction (MCA404)

b) Cloud Computing (MCA405)

c) ERP & Supply Chain Management (MCA406)

Department Elective 2:

a) Information Storage & Management (MCA407)

b) Artificial Intelligence & Neural Networks (MCA408)

c) Software Testing Methodologies (MCA409)

COURSE STRUCTURE

Master of Computer Applications

..III.. Year ..V..Semester

S. No.	Subject Code	Subject Title	Periods per week			C	Scheme of Examination Maximum Marks		
			T	P	D		Int.	Ext.	Total
1.	MCA501	Cryptography & Network Security	5	-	-	4	40	60	100
2.	MCA502	OOAD through UML	5	-	-	4	40	60	100
3.	MCA503	Visual Programming	5	-	-	4	40	60	100
4.		Elective - 3	5	-	-	4	40	60	100
5.		Elective - 4	5	-	-	4	40	60	100
6.	MCA510	OOAD through UML Lab	-	3	-	2	50	50	100
7.	MCA511	Visual Programming Lab	-	3	-	2	50	50	100
		Total	**25**	**6**		**24**	**300**	**400**	**700**

T- THEORY P – PRACTICAL D- DRAWING C – CREDITS Int. – INTERNAL Ext. - EXTERNAL

LIST OF ELECTIVES

Department Elective 3:

a) E-Commerce (MCA504)

b) Software Design Methodologies (MCA505)

c) Design Patterns (MCA506)

Department Elective 4:

a) Mobile Application Development (MCA507)

b) Software Project Management (MCA508)

c) Data Sciences (MCA509)

COURSE STRUCTURE

Master of Computer Applications

..III.. Year ..VI..Semester

S. No.	Subject Code	Subject Title	Periods per week			C	Scheme of Examination Maximum Marks		
			T	P	D		Int.	Ext.	Total
1.	MCA601	Seminar	2	-	-	2	50	-	50
2.	MCA602	Dissertation / Thesis /VivaVoce	-	8	-	10	-	150	150
		Total	**2**	**8**		**12**	**50**	**150**	**200**

T- THEORY P – PRACTICAL D- DRAWING C – CREDITS Int. – INTERNAL Ext. – EXTERNAL

I Year MCA – I Semester

MCA101 C Programming and Data Structures

Course Objective: ***This course is designed provide a comprehensive study of the Formulating algorithmic solutions to problems and implementing algorithms in C.***

Unit -I

Objective: Notion of Operation of CPU, Notion of an algorithm and computational procedure, editing and executing programs in IDE's.

Introduction to Computers, H/W and S/W concepts, Algorithm, pseudo code, flowchart, program development steps, Introduction to various IDE's and their use in C program development, structure of C program, A Simple C program, identifiers, basic data types and sizes, Constants, variables, arithmetic, relational and logical operators, increment and decrement operators, conditional operator, bit-wise operators, assignment operators, expressions, type conversions, conditional expressions, precedence and order of evaluation. Control structures such as if, go to, labels, and switch statements.

Unit -II

Objective: Understanding branching, iteration and data representation using arrays.

Loops- while, do-while and for statements, break, continue, Arrays - concepts, declaration, definition, accessing elements, storing elements, Strings and string manipulations, 1- D arrays other than strings, 2-D character arrays – 2-D arrays other than character arrays – Multidimensional arrays.

Unit -III

Objective: Modular programming and recursive solution formulation. Understanding pointers and dynamic memory allocation.

Functions:

Basics, parameter passing, storage classes- extern, auto, register, static, scope rules, block structure, user defined functions, standard library functions, recursive functions, header files, C pre-processor. Passing 1-D arrays, 2-D arrays, and functions.

Pointers:

Concepts, initialization of pointer variables, pointers and function arguments, passing by address –dangling memory, Character pointers and functions, pointers to pointers, pointers and multidimensional arrays, dynamic memory management's functions, command line arguments.

Unit - IV

Objective: Understanding miscellaneous aspects of c. Comprehension of file operations and Data Structures using sorting techniques.

Derived types:

Structures- declaration, definition and initialization of structures, accessing structures, nested structures, arrays of structures, structures and functions, pointers to structures, self referential structures, unions, typedef, bit-fields, Input and output – concept of a file, text files

and binary files, Formatted I/o, file I/o operations.

Data Structures:
Introduction to Data Structures – Time Complexity – Space Complexity – Pattern matching – naive method – Robin Karp Algorithm - Searching – Linear and binary search methods, sorting – Bubble sort, selection sort, Insertion sort, Quick sort, merge sort.

Unit -V
Objective: Understand about linear & non-linear data structures and their applications.

Single linked lists, doubly linked lists, circular list, representing stacks and queues inC using arrays and linked lists, infix to post fix conversion, postfix expression evaluation. Trees- Binary tress, terminology, representation, traversals, Graphs - terminology, representation, graph traversals (dfs & bfs) – Warshalls – Dijkstra – Kruskal – Prims Algorithms. Only Algorithms.

Text Books:

1. C and Data Structures: A snapshot oriented treatise using live Engineering examples, N B Venkateswarlu, E. V Prasad, S Chand & Co.
2. Computer science, A structured programming approach using C, B.A. Frozen and R.F. Gilberg, Third edition, Thomson.

References:

1. Fundamentals of Data Structures in C , Horowitz, Sahni, Anderson-Freed, 2nd , Universities Press, 2008.
2. Classic Data Structures, Samanta, 2nd, PHI, 2009.
3. The C Programming Language, B.W. Kernighan, Dennis M.Ritchie, P H I / Pearson.
4. C Programming with problem solving, J.A. Jones & K. Harrow, Dramatic Press
5. Data Structures Using C, A.S.Tanenbaum, Y. Langsam, and M.J. Augenstein, PHI/Pearson.
6. Programming in C, Stephen G. Kochan, III Edition, Pearson .
7. Data Structures and Program Design in C, R.Kruse,, Tondo, Leung, Shashi M, 2nd Edition, Pearson.
8. Data Structures and Algorithms, Aho, Hopcroft, Ullman, Pearson ,2006
9. C and Data Structures, Ashok N.Kamthane, Pearson.
10. C Programming and Data Structures, E Balaguruswamy, TMH, 2008.

Web Resources:

1. http://jntubitss.blogspot.in/2013/02/computer-programming-and-data.html
2. http://forum.jntuworld.com/showthread.php?5544-Computer-Programming-amp-Data-Structures-%28CPDS-C-amp-DS%29-Study-Materials-Notes
3. http://www.ajntuworld.in/2-1-cse-data-structures-pdf/
4. http://downloads.ziddu.com/download/7589002/CNotes_ww8.pdf.html/eng

I Year MCA – I Semester

MCA102 Digital Logic and Computer Organization

Course objective: ***Understand the architecture of a modern computer with its various processing units. Also the performance measurement of the computer system. In addition to this the memory system of computer.***

Unit -I
Objective: Understand the architecture of modern computer.

Digital Components and Data Representation:
Introduction, Numbering Systems, Decimal to Binary Conversion, Binary Coded Decimal Numbers, Weighted Codes, Self-Complementing Codes, Cyclic Codes, Error Detecting Codes, Error Correcting Codes, Hamming Code for Error Correction, Alphanumeric Codes, ASCII Code.

Boolean Algebra and Logic Gates:
Introduction, Postulates of Boolean Algebra, Basic Theorems of Boolean Algebra, Duality Principle, Theorems, Precedence of operators, Venn Diagram , Boolean Functions and Truth Tables, Canonical Forms for Boolean Functions, Binary Operators and Logic Gates, Simplifying Boolean Expressions, Veitch-Karnaugh Map Method, Four Variable Karnaugh Map, Incompletely Specified Function, Quine-McCluskey Procedure.

Unit -II
Objective: Understand about data storage units and their design.

Digital logic circuits:
Combinatorial Switching Circuits: Introduction, Combinatorial Circuit Design Procedure, Integrated NAND-NOR Gates, CMOS Transistor Gates, Realization of Boolean Expressions Using NAND/NOR Gates, Combinatorial Circuits Commonly Used in Digital Systems, Design of Combinatorial Circuits with Multiplexers, Programmable Logic Devices, Realization with FPLAs, Realization with PALs.

Sequential Switching Circuits:
Types, Flip-Flops, Counters, Modelling Sequential Circuits – FSM. Synthesis of synchronous, Binary counters.

Unit -III
Objective: Understand of a computer performs arithmetic operations of numbers.

Arithmetic and Logic Unit:
Introduction, Binary Addition, Binary Subtraction, Complement, Representation of Numbers, Addition/Subtraction of Numbers in 1's Complement Notation, addition/Subtraction of Numbers in Two's Complement Notation, Binary Multiplication, Multiplication of signed Numbers, Binary division, Integer Representation, Floating Point Representation of Numbers, Binary Floating Point Numbers, IEEE Standard Floating Point Representation, Floating Point addition/Subtraction, Floating Point Multiplication, Floating Point Division, Floating Point Arithmetic Operations, Logic Circuits for Addition/Subtraction, Half- and Full-Adder Using Gates, A Four-bit Adder, MSI arithmetic Logic Unit, A Combinatorial Circuit for Multiplication.

Unit -IV
Objective: Understand CPU Instruction set format and addressing modes and micro program control.

Central Processing Unit:
Learning Goals, Introduction, Operation Code Encoding and Decoding, Instruction Set and

Instruction Formats, Addressing Modes, Register Sets, Clocks and Timing, CPU Buses, Dataflow, Data Paths and Microprogramming, Control Flow.

Micro programmed Control:
Control Memory, Address Sequencing, Conditional Branching, Mapping of Instruction, Subroutines, Micro program Example, Computer Configuration, Microinstruction Format, Symbolic Microinstructions, The Fetch Routine, Symbolic Micro program, Binary Micro program , Design of Control Unit, Micro program Sequencer.

Unit -V
Objective: To gain knowledge Memory and IO organization.

Memory Organization:
Introduction, Memory hierarchy, Dynamic Random Access Memory, Error Detection and Correction in Memories, Read Only Memory, Dual-Ported RAM, Enhancing Speed and Capacity of Memories, Program Behaviour and Locality Principle, Cache in Memory Organization, Design and Performance of Cache Memory System.

Input-Output Organization:
Introduction, device Interfacing, Overview of I/O Methods, Program Controlled Data Transfer, Interrupt Structures, Single level Interrupt Processing, Handling Multiple Interrupts, Interrupt Controlled data Transfer,DMA Based Data Transfer, Input/output (I/O) Processors, Bus Structure, Structure of a Bus, Types of Bus, Bus Transaction Type , Serial Data Communication, Asynchronous Serial data communication.

Text Books:

1. Digital Logic and Computer Organization, Rajaraman, Radhakrishnan, PHI, 2006
2. Digital Logic Design, Moriss Mano, PHI
3. Computer System Architecture, 3rd ed ., M. Morris Mano, PHI, 1994

References:

1.Computer Organization, 5th ed., Hamacher, Vranesic and Zaky, TMH,2002
2.Computer System Organization & Architecture, John D. Carpinelli,Pearson, 2008
3.Computer System Organization, Naresh Jotwani, TMH, 2009
4.Computer Organization & Architecture: Designing for Performance, 7th ed., William Stallings, PHI, 2006

Web Resources:

1. http://jfufiles.com/my-jfufiles-downloads/1581532167
2. http://www.slideshare.net/vanithachandru/computer-organization-logic-gates- bool
3. http://home.adelphi.edu/~siegfried/cs371/37113.pdf

I Year MCA – I Semester

MCA103 Discrete Mathematical Structures and Graph Theory

Course Objective: ***Read and understand definitions and proofs using basic discrete Mathematics.***

Unit -I
Objective: Understand Mathematical logic and Predicate calculus.

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 of premises and indirect method of proof, Automatic Theorem Proving.

Predicate Calculus:
Predicates, statement functions, variables and quantifiers, predicate formulas, free & bound variables, universe of discourse, inference theory of predicate calculus.

Unit -II
Objective: To gain knowledge about Set theory & relations and Functions.

Set Theory & Relations:
Introduction, Relations and ordering, Properties of binary Relations, Equivalence, Compatibility Relations, Partial ordering, Hasse diagram.

Functions:
Composition of functions, Inverse Function, Recursive Functions, Lattice and its Properties, Pigeon hole Principles and its application.

Algebraic Structures:
Algebraic systems, Examples and general properties, Semi groups and monoids, groups, sub groups, Definitions, Examples, homomorphism, Isomorphism and related problems.

Unit -III
Objective: Understand Elementary Combinatory usage in life.

Elementary Combinatory:
Basis of counting, Enumeration of Combinations & Permutations, Enumerating of Combinations & Permutations with repetitions and constrained repetitions, Binomial Coefficients, Binomial Multinomial theorems, principles of Inclusion – Exclusion.

Unit -IV
Objective: Understand recurrence relations usage in solve recursive process.

Recurrence Relations:
Generating Function of Sequences, Calculating Coefficient of generating functions, Recurrence relations, Solving recurrence relation by substitution and Generating functions, The method of Characteristic roots, Solution of Inhomogeneous Recurrence Relation.

Unit -V
Objective: Understand Graph Theory and it's applications.

Graph Theory:
Representation of Graph, Spanning Trees, BFS, DFS, Kruskals Algorithm, Binary trees,

PlanarGraphs, Graph Theory and Applications, Basic Concepts, Isomorphism and Sub graphs, Multi graphs and Euler circuits,Hamiltonian graphs, Chromatic Numbers.

Text Books:

1. Discrete Mathematical Structures with Applications to computer science J.P Tremblery, R.Manohar, TMH
2. Discrete Mathematical for computer Scientists & Mathematicians " J.L. Molt, A.Kandel ,T.P.Baker, PHI

References:

1. Elements of Discrete Mathematics, C L Liu, D P Mohanpatra,TMH
2. Discrete Mathematics, Schaum's Outlines,Lipschutz,Lipson TMH.
3. Discrete Mathematical Structures, Kolman, Busby, Ross, 6th ed., PHI, 2009
4. Discrete Mathematics, Johnsonbaugh, 6th ed., Pearson, 2005
5. Discrete Mathematics, Malik, Sen, 6th ed., Cengage Learning, 2004
6. Discrete Mathematics for computer science, Bogart, Stein and Drysdale,Springer,2005
7. Discrete Mathematics and Combinatorics, Sengadir, Pearson, 2009
8. Discrete and Combinatorial Mathematics, Grimaldi, Ramana, 5th ed., Pearson.2006
9. Mathematical Foundations of Computer Science, Rajendra Prasad, Rama Rao ET al.,USP, 2009
10. Discrete Mathematics, J K Sharma, 2nd ed., Macmillan, 2005
11. Discrete Mathematics with Combinatorics and Graph Theory, Santha, Cengage Learning, 2009
12. Applied Discrete Structures For Computer Science, Alan Doerr, Levassure, GP, 2005
13. Discrete Mathematics with Applications, Koshy, Elsevier,2006.
14. Discrete Mathematics and its Applications, Rosen, 5th ed, T M Graw-Hill ed, 2006.
15. Discrete Mathematics for Computer Science, Gary Haggard, John Schlipf, Sue Whiteside's, Cengage, 2006.
16. Discrete Mathematical, Kevin Ferland, Cengage, 2008.
17. Discrete Mathematical Structures, Jayant Ganguly, Sanguine, 2007.

Web Resources:

1. http://forum.jntuworld.com/showthread.php?6366-Mathematical-Foundations- of-Computer-Science--%28MFCS%29-Notes-Material
2. http://jkdirectory.blogspot.in/2010/11/mfcs.html
3. http://cs.bme.hu/fcs/graphtheory.pdf
4. http://math.tut.fi/~ruohonen/GT_English.pdf

I Year MCA – I Semester

MCA104 Probability and Statistical Applications

Course Objective: ***This subject gives the knowledge about Probability and Statistical Applications***

Unit -I
Objective: Understand baye's theorem, total probability theorem.

Probability Theory:
Sample spaces Events & Probability; Discrete Probability; Union , intersection and compliments of events; Conditional probability ;Baye's theorem.

Unit -II
Objective: Understand random variables and distributions.

Random variables and distribution:
Random variables Discrete Probability Distributions, Continuous probability distribution, Binomial, Poisson, uniform, Exponential, Normal.

Unit -III
Objective: The knowledge of testing of hypothesis for all size of samples.

Expectations and higher order moments – Moment Generating Function, Characteristic functions – Laws on large numbers – Weak Laws and strong laws of large numbers. Central limit theorem and other limit theorems.

Sampling Distribution:
Populations and samples - Sampling distributions of mean (σ known and unknown) proportions, sums and differences. Statistics based on Normal, Student's t and F distributions.

Unit -IV
Objective: This unit also gives the knowledge of finding out the coefficient of correlation, Regression line equations and queuing theory.

Tests of significance –Z-test, t-test, F-test, χ^2test.
Linear correlation coefficient linear regression;Non Linear regression least square fit; polynomial and Curve fittings.

Time Series and Forecasting:
Moving averages, Smoothening of curves Forecasting models and methods, Statistical Quality Control Methods-bar charts p-charts etc.

Unit -V
Objective: This unit also gives the knowledge of queuing theory.

Queuing theory – Markov Chains – Introduction to Queuing systems – Elements of a queuing model – Exponential distribution – Pure birth and death models. Generalized Poisson Queuing model – Specialized Poisson Queues.

Text Books:

1. Probability, Statistics and Random Processes Dr.K.Murugesan & P.Gurusamy By Anuradha Agencies, Deepti Publications.
2. Probability, Statistics and Random Processes, T.Veerarajan, TMH, India

References:

1. Probability and Statistics for Engineers: Miller and Freund, PHI.
2. Probability, Statistics and Queuing Theory Applications, 2nd ed, Trivedi, John Wiley and Sons.

Web Resources:

1. http://www.ziddu.com/download/15701672/PSNotes.rar.html
2. http://www.mediafire.com/?z217j9hnp3m2n8s
3. http://forum.jntuworld.com/showthread.php?17173-Probability-amp-Statistics-Important-Questions
4. http://www.ziddu.com/download/1463908/probability.pdf.html
5. http://forum.jntuworld.com/showthread.php?18027-Probability-and-Statistics-Notes-Study-Materials- For-JNTU-HYD-JNTU-KAKINADA-amp-JNTU-ANANTAPUR

I Year MCA – I Semester

MCA105 Accounting and Financial Management

Course Objective: ***This course is an introduction to financial accounting. The preparation and use of financial statements examined.***

Unit -I
Objective: To gain knowledge about Accounting principles, Preparation of trail balance and Final accounts.

Accounting Generally Accepted Accounting Principles (GAAP) & Accounting standards, Characteristics and limitations of single entry system, double entry system of accounting, introduction of basis books of accounts, ledgers.

Preparation of trail balance – Final accounts – company final accounts – Users of Accounting Information, Role of Accountant in modern Organizations.

Unit -II
Objective: Understand financial management, Ratio analysis and Fund flow analysis.

Financial Management – meaning and scope, role, objectives of time value of money – over vitalization – under capitalization – profit maximization – wealth maximization – EPS maximization. Ration Analysis - advantages - limitations - Fund flow analysis – meaning, importance, preparation and interpretation of Funds flow and cash flow statements – statements of changes in working capital.

Unit -III
Objective: Understand Costing, Marginal costing and Break –even analysis.

Costing – nature and importance and basic principles. Elements of cost – Absorption costing Vs. Marginal costing – Financial accounting vs. cost accounting vs. management accounting.

Marginal Costing and Break – Even Analysis :
Nature, scope and importance – practical applications of marginal costing, limitation and importance of cost – volume, profit analysis, short run decisions.

Unit -IV
Objective: To gain knowledge about Standard costing and budgeting variance and budget.

Standard Costing and Budgeting :
Nature, scope and computation and analysis – materials variance, labor variance and sales variance – cash budget, sales - budget – flexible Budgets, master budgets.

Unit -V
Objective: Understand computerized accounting system.

Introduction to Computerized Accounting System:
Coding logic and codes, master files, transaction files, introduction documents used for data collection, processing of different files and Outputs obtained.

Text Books:

1. Accounting for Managers, P. Vijaya Kumar, and Himalaya Publications.

References:

1. Accounting for Management. Vijaya Kumar.TMH.
2. Financial Accounting, S.N Maheswari and S.K. Maheswari, Vikas.
3. Financial Accounting, A. Mukherjee and M. Heneef, TMH.
4. Basic Financial Accounting for MPanagement, Ambaresh Gupta, Pearson.
5. Accounts and Finance for Non accounts, Chatterjee, D.K. Himalaya.
6. Financial Analysis and Accounting, P. Premchand Babu and M. Madam Mohan,Himalaya.
7. Essential of Financial Accounting, Ashish, K and Ballacharya, PHI.
8. Guide to Financial Management, John Tannent, Viva.

Web Resources:

1. http://archive.mu.ac.in/myweb_test/MCA%20study%20material/M.C.A.%20(Sem%20%20II)%20Accounting%20and%20Financial%20Management.pdf
2. http://www.slideshare.net/BabasabPatil/financial-and-management-accounting-notes-mba-bk
3. http://education.svtuition.org/2011/08/financial-management-notes.html
4. http://www.icai.org/post.html?post_id=6081
5. http://www.ziddu.com/downloadlink/1456128/MEFA_Most_Important_Questions.doc

I Year MCA – I Semester

MCA106 English Language Communication Skills Lab

Course Objective: *The language lab focuses computer-aided multi-media instruction and Language acquisition to achieve the following targets:*

1. *To expose the students to a variety of self-instructional, learner-friendly modes of language learning.*
2. *To help the students cultivate the habit of reading passages from the computer monitor, thus providing them with the required facility to face computer-based competitive exams such GRE, TOEFL, GMAT etc.*
3. *To enable them to learn better pronunciation through stress on word accent, intonation, and rhythm.*
4. *To train them to use language effectively to face interviews, group discussions, public speaking.*
5. *To initiate them into greater use of the computer in resume preparation, report writing, format- making etc.*

However, depending upon the available of infrastructure and budget, the above targets can also be achieved by procuring the minimum required equipment suggested for the establishment of Conventional Lab the details of which are given below. The lab should cater to the needs of the students to build up their confidence to help them develop leadership qualities through their communicative competence.

English Language Laboratory Practice

1. Introduction to Phonetics. 2. Introduction to Vowels and Consonants and associated Phonetic symbols. 3. Introduction to Accent, Intonation and Rhythm. 4. Situational Dialogues/Role Play. 5. Debate 6. Public Speaking. 7. Group Discussions 8. Facing Interviews 9. Resume preparation 10. e-Correspondence

MODULE	TOPICS/SUB-TOPICS	LAB SESSION
1.	INTRODUCTION TOPHONETICS -Vowels, -Consonants, -Diphthongs INTRODUCTION TO STRESS & INTONATION -Articulation, -Respiration, -Phonation	3
2.	GROUP DISCUSSIONS FACING INTERVIEWS	4
3	SITUATIONAL/DIALOGUE/ ROLE PLAY RESUME PREPARATION	2
4	PUBLIC SPEAKING, DEBATE	2
5	GRE,TOEFL,GMAT MODELS, e-CORRESPONDENCE	3

Suggested Software for Lab classes:

Cambridge Advanced Learners' Dictionary with exercises
The Rosetta stone English Library
Clarity Pronunciation Power
Mastering English in Vocabulary, Grammar, Spellings, Composition
Dorling Kindersley series of Grammar, Punctuation, Composition etc.
Oxford Advanced Learner's Compass, 7th Edition
Language in Use, Foundation Books Pvt Ltd
Learning to Speak English - 4 CDs
Microsoft Encarta
Murphy's English Grammar, Cambridge
Time series of IQ Test, Brain-teasers, Aptitude Test etc.
English in Mind, Herbert Puchta and Jeff Stranks with Meredith Levy, Cambridge

Reference Books:

1. The Human Touch: personal Skills for Professional Success – by Debra Paul.
2. The Definitive Book of body Language – by Allan Pease, Barbara Pease.
3. How to Face Interviews – by Clive Fletcher.
4. The 7 Habits of Highly Effective People – by StephenCovery.
5. The Google Resume: How to Prepare of a Career and Land a Job at Apple, Microsoft.
6. Good English –by G.H Vallns
7. Better English – G.H Vallins
8. Best English – G.H. Vallins
9. How to talk to anyone: 92 little tricks for big success in Relationships by Leli Lowndes.
10. The leader in you - by Dale Carnagie
11. 250 Job Interview Questions You 'I most likely be asked – by Peter veluki, Peter Verki.
12. Contemporary English Grammar, structures and Composition - by David Green.

I Year MCA – I Semester

MCA107 Computer Programming Data Structures Lab

Course Objective: ***To learn/strengthen a programming language like C, To learn problem solving Techniques To introduce the student to simple linear and non linear data structure such as lists, stacks, queues, etc.,.***

Recommended Systems/Software Requirements:

- Intel based desktop PC, A N S I C Compiler with Supporting Editors, IDE's such as Turbo C, Bloodshed C.

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.
d) Write a program which checks a given integer is Fibonacci number or not.

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.
c) Write a C program to implement Newton Raphson method for a quadratic equation.
d) Write a C program to implement Newton Raphson method for a general Purpose algebraic equation.

Exercise 3

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.
iii) To solve Towers of Hanoi problem. iv) Write program to calculate probability of head/tail by generating random numbers using random () function.

Exercise 4

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 form the user, performs the operation and then prints the result. (Consider the operators +,-,*, /, % and use Switch Statement)

Exercise 5

a) Write a C program to find both the larges and smallest number 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
iii) Checking symmetricity of a square matrix. iv) Calculating transpose of a matrix in-place manner.

Exercise 6

a) Write a C program that uses functions to perform the following operations:
i) To insert a sub-string in to 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/ 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: $1+x+x^2+x^3+\ldots\ldots\ldots\ldots+x^n$
For example: if n are 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 using Structure:
i) Reading a complex number ii) Writing a complex number
iii) Addition of two complex numbers iv) Multiplication of two complex numbers

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 that uses functions to perform the following operations on singly linked list.

i) Creation ii) Insertion iii) Deletion iv) Traversal

b) Adding two large integers which are represented in linked list fashion.

Exercise 14

Write a C program that uses functions to perform the following operations on doubly linked list.:
i) Creation ii) Insertion iii) Deletion iv) Traversal in both ways

Exercise 15

a.) Write C programs that implement stack (its operations) using:
i) Arrays i i) Pointers iii) linked list.

Exercise 16

a. Write C programs that implement Queue (its operations) using:
i) Arrays i i) Pointers i i i) linked lists

Exercise 17

Write a C program that uses Stack operations to perform the following:
i) Converting infix expression into postfix expression ii) Evaluating the postfix expression

Exercise 18

a. Write a C program that uses functions to perform the following:
i) Creating a Binary Tree of integers
ii) Traversing the above binary tree in preorder, inorder and postorder
b. Program to check balance property of a tree.
c. Program to check for its strictness.

Exercise 19

Write C programs that use both recursive and non recursive functions to perform the following searching operations:
for a Key value in a given list of integers : i) Linear search ii) Binary search

Exercise 20

Write C programs that implement the following sorting methods to sort a given list of integers in ascending order:
i) Bubble sort ii) Quick sort

Exercise 21

a. Write C programs that implement the following sorting methods to sort a given list of integers

in ascending order:

i) Insertion sort i i) Bubble sort

b. Recursive implementation of sorting algorithms.

Exercise 22

Write C programs to implement the Lagrange interpolation and Newton- Gregory forward interpolation.

Exercise 23

a. Program to calculate mean and standard deviation of a population.
b. Write C programs to implement the linear regression and polynomial regression algorithms.

Exercise 24

a.Write C programs to implement Trapezoidal and Simpson methods. and b) Program for Calculating pi value.

References:

1. Digital Fundamentals, Floyd, Jain, 8th ed , Pearson
2. Digital Logic and Computer Organization, Rajaraman, Radhakrishnan, PHI, 2006

I Year MCA– I Semester

MCA108 Digital Logic and Computer Systems Organization (DLCSO) Lab

Course Objective: ***To learn each and every thing related computer operations, hardware parts functioning and testing memory & storage components.***

Exercise 1

Boolean algebra: Theorems and logical guides, verification of truth tables.

Exercise 2

Realization of Boolean expressions; Using (i) AND – OR-NOT Gates (ii) NAND Gates (iii) NOR Gates.

Exercise 3

Latches Flip – Flops: RS, JK, T, D, Master –Slave FF, Edge – Triggered Flip – Flops.

Exercise 4

Counters: Binary Counter, Synchronous/Asynchronous Binary Counter, Ripple Counter, Decade Counter, Up/Down Counter.

Exercise 5

Modulo Counter: Modulo - 5, Modulo – 10.

Exercise 6

Adders / Sub tractors: Half Adder, Full Adder, 1 's and 2's complement addition.

Exercise 7

Multiplexers/ Data Selector: 2- input and 8- input, Demultiplexers , Logic Function Generator.

Exercise 8

Decoders and Encoders.

Exercise 9

BCD adders and Comparators.

Exercise 10

Registers: Basic Shift Register (SR), SI/SO SR, SI/PO SR, PI/SO SR, PI/PO SR.

Exercise 11

Johnson Counter, Sequence Generator, Parity Generators/ Checkers.

Exercise 12

Code Converters : Decimal –to-Binary, Binary – to – Decimal, Decimal – to- Hexa Decimal, BCD- to –Decimal, Binary – to- gray, gray- to –Binary.

Exercise 13

Buffers / Derivers: Open; collector Buffers.

Exercise 14

Gates: CMOS / NMOS/TTL – Basic Operational Characteristics and parameters.

Exercise 15

RAM, ROM, PROM, EPROM – Testing Memory Chips.

References:

1. Digital Fundamentals, Floyd & Jain, Pearson, 2005.
2. Digital Logic and Computer Organization, Rajaraman, Radhakrishnan, PHI,2006

I Year MCA – II Semester

MCA201 Object Oriented Programming with C++

Course Objective: ***To impart knowledge on object oriented programming with the help of C++ to impart the knowledge on programming concepts and advanced C++ concepts to improve the productivity with emphasis on new methods and techniques.***

Unit – I

Objective: To introduce basic concepts of C++ and be familiar with writing recursive methods.

Introduction:

Overview of C++, Sample C++ program, Different data types, operators, expressions, and statements, arrays and strings, pointers & function components, recursive functions, user-defined types, function overloading, inline functions.

Classes & Objects–I: Classes, Scope resolution operator, passing objects as arguments, returning objects, and object assignment.

Unit- II

Objective: Explanation of concepts like Functions, classes and overloading with the help of C++ and the use of functions

Classes & Objects–II: Constructors, Destructors, friend functions, Parameterized constructors, Static data members, Functions, Arrays of objects, Pointers to objects, this pointer, and reference parameter, Dynamic allocation of objects, Copy constructors, Operator overloading using friend functions such as +, -, pre-increment, post-increment, overloading <<.

Unit – III

Objective: Introduce concepts of inheritance and the implementation of those concepts with the help of C++

Templates:

Generic functions and Generic classes

Inheritance :

Base Class, Inheritance and protected members, Protected base class inheritance, Inheriting multiple base classes, Constructors, Destructors and Inheritance, Passing parameters to base class constructors, Granting access, Virtual base classes.

Unit – IV

Objective: Explanation of polymorphism and the methods included in it and the implementation of abstract classes and virtual functions.

Virtual functions, Polymorphism:
Virtual function, calling a Virtual function through a base class reference, Virtual attribute is inherited, Virtual functions are hierarchical, pure virtual functions, Abstract classes, Using virtual functions, Early and late binding.

Unit – V
Objective: Deriving the exception handling mechanisms and introduce file concepts which are used to read, write, open and close the files.

Exception Handling, I/O System Basics, File I/0:
Exception handling fundamentals, Exception handling options. C++ stream classes, Formatted I/O, C++ File I/O: <fstream> and the File classes, Opening and closing a file, Reading and writing text files.

Text Book:
1. Herbert Schildt: The Complete Reference C++, 4th Edition, Tata McGraw Hill, 2011.

References:
1. Stanley B.Lippmann, Josee Lajoie: C++ Primer, 4thEdition, Addison Wesley, 2012.
2. Paul J Deitel, Harvey M Deitel: C++ for Programmers, Pearson Education, 2009.
3. K R Venugopal, Rajkumar Buyya, T Ravi Shankar: Mastering C++, 1st edition, Tata McGraw Hill, 2011.

Web Resources:
1. www.atilim.edu.tr/~mcs215/Lecture%20Notes/
2 https://atomicobject.com/resources/handbook-of.../c-plus-plus-language
3 https://realm.io/.../altconf-peter-steinberger-**objective-c++**-what-could-po..
4 https://developer.apple.com/library/mac/.../**Conceptual**/.../tags/tags.html

I Year MCA – II Semester

MCA202 UNIX & Network Programming

Course Objective: ***Understand what an operating system is and how UNIX fulfills that role. Introduce UNIX programming techniques. Discuss the role of shell in operating system***

Unit – 1

Objective: Explanation of shell file concepts and shell programming. Teach advanced c concepts and techniques in unix environment.

Review of UNIX Utilities and Shell Programming:

File handling utilities, security by file permissions, process utilities, disk utilities, networking commands, backup utilities, text processing utilities, Working with the Bourne shell, What is a shell, shell responsibilities, pipes and input redirection, output redirection, here documents Shell as a programming language, shell meta characters, shell variables, shell commands, the environment, control structures, shell script examples.

Unit -II

Objective: Know the significance of the important directories of the UNIX file system from a functional point of view.

UNIX Files:

Unix file structure, directories, files and devices, System calls, library functions, low level file access,usage of open, creat, read, write, close, lseek, stat, fstat, octl, umask, dup, dup2. The standard I/O (fopen, fclose,fflush, fseek, fgetc, getc, getchar, fputc, putc, putchar, fgets, gets), formatted I/O, stream errors, streams and filedescriptors, file and directory maintenance (chmod, chown, unlink, link, symlink, mkdir,rmdir,chdir, getcwd),handling system calls (opendir, readdir, closedir, rewinddir, seekdir, telldir)

Unit -III

Objective: An overview of the kernels role in process management. Examine process attributes and inheritance mechanism.

UNIX Process Threads and Signals:

What is process, process structure, starting new process, waiting for a process, zombie process, process control, process identifiers, system call interface for process management, fork, vfork, exit, wait, waitpid, exec, system, Threads, Thread creation, waiting for a thread to terminate, thread synchronization, variables, cancelling a thread, threads vs. processes. Signals, Signal functions, unreliable signals, interrupted system calls, kill and raise functions, alarm, pause functions, abort, sleep functions.

Unit- IV

Objective: Overview of data management in unix environment and detail about the inter process communication. Explain about the concept of message queue.

Data Management:
Management Memory (simple memory allocation, freeing memory) file and record locking (creating lock files, locking regions, use of read/ write locking, competing locks, other commands, deadlocks). Inter process Communication-Introduction to IPC, IPC between processes on a single computer system, IPC between processes on different systems, pipes, FIFOs. streams and messages, namespaces, introduction to three types of IPC (system-V)- message queues, semaphores and shared memory.

Message Queues:
Permission issues, Access permission modes, message structure, working message queues, Unix kernel support for messages.

Unit- V
Objective: Overview of the semaphores and shared memory .

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, working with a shared memory segment, 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.

Text Books:

1. Unix Concepts and Applications, 3/e, Sumitabha Das, TMH
2. Advanced Unix Programming, N B Venkateswarlu, BSP

References:

1. Unix and shell Programming, Sumitabha Das, TMH
2. A Beginner's Guide to Unix, N.P.Gopalan, B.Sivaselva, PHI
3. Unix Shell Programming, Stephen G.Kochan, Patrick Wood, 3/e, Pearson
4. Unix and shell Programming, N B Venkateswarlu, Reem, New Delhi
5. Unix Programming, Kumar Saurabh, Wiley,India

Web Resources:

1. https://en.wikipedia.org/wiki/W._Richard_Stevens
2. https://www.cs.drexel.edu/~kschmidt/CS265/Lectures/.../**shellScripting**.pp.
3. www.cs.toronto.edu/~yganjali/resources/.../H03--
4. https://msdn.microsoft.com/en-us/library/ms973860.aspx
5. www.cs.nyu.edu/courses/fall05/G22.2631-001/conron.doc

I Year MCA – II Semester

MCA203 Operating Systems

Course Objective: ***Understand the concepts, structures and mechanisms of operating systems. The emphasis of the course will be placed on understanding how various elements that underlie operating system interact and provides services for execution of application software.***

Unit -I

Objective: Provides an overview of computer architecture and organization, with emphasis on topics related to operating system design

Introduction to Operating Systems:

OS structure and strategies, Evolution of operating systems-simple, batch, Multi programmed, time shared, personal computer, parallel, distributed systems, real time systems. Operating-system Services, Operating-system Interface, System calls, Types of System calls, System programs, Operating-system Design and Implementation.

Unit- II

Objective: Presents a detailed analysis of processes, multithreading. Provides a discussion of various approaches to process scheduling

Process Management:

Processes: Process Concept, Process Scheduling, Operations on Processes , Inter process Communication, Examples of IPC Systems, Communication in Client-Server systems Threads: Overview, Multithreading Models.

CPU Scheduling:

Basic concepts, Scheduling Criteria, Scheduling Algorithms, Multiple-Processor Scheduling, Thread Scheduling.

Process Synchronization:

Background, the Critical- section problem, Peterson's solution, Synchronization Hardware, Semaphores, Classic problems of Synchronization, Monitors, Atomic Transactions.

Unit- III:

Objective: Provides a comprehensive survey of techniques for memory management, including virtual memory

Memory management:

Main memory: Swapping, Contiguous memory Allocation, Paging, Structure of the Page table, Segmentation

Virtual memory:

Background, Demand paging, copy-on-Write, Page Replacement, Allocation of frames, Thrashing, Memory-Mapped Files.

Unit -IV

Objective: Master concepts of file system interface and implementation, disk management

File-system Interface:

Concept, Access Methods, Directory structure, File-system Mounting, File sharing, Protection. File-system Implementation: File-system Structure, Implementation, Directory Implementation, Allocation Methods, Free- Space Management, Efficiency and Performance, Recovery, Log-Structured File systems,NFS.

Device Management:

Disk Structure, Disk Attachment, Disk Scheduling, Disk Management, Swap Space Management, RAID structure. Stable storage Implementation. I/O System: I/O hardware, Application I/O Interface, Kernel I/O Subsystem, Transforming I/O request to hardware operation, STREAMS.

Unit -V

Objective: Be familiar with protection and security mechanisms

Deadlocks:

Deadlocks: Necessary conditions, resource allocation graph, methods for handling deadlocks, preventions, avoidance, detection and recovery.

Protection:

Goals of Protection, Principles of protection, Domain of Protection.

Security:

The Security Problem, Program Threads, System and Network Threats, Cryptography as a security tool, User Authentication, Firewall to protect systems and Networks.

Textbooks:

1. Abraham Silberschatz, Peter B. Galvin, Greg Gagne, Operating System Concepts, Wiley India, 2006.

References:

1. Operating systems,6/E,William stallings, PHI/Pearson.
2. Operating systems 3/e,Dietal ,Dietal,Pearson.
3. An introduction to Operating systems, Concepts and practice, Pramod Chandra P.Bhat,PHI
4. Operating systems, Haldar, Aravind, Pearson.
5. Operating systems, PAL Choudhury, PHI.
6. Operating systems: design and Implementation, 3/e,Tanenbaum, Woodhull.

Web Resources:

1. https://www.quora.com/What-is-the-best-book-on-operating-systems
2. https://en.wikipedia.org/wiki/Memory_management_
3. www.tutorialspoint.com/operating_system/os_memory_management.
4. www.personal.kent.edu/~rmuhamma/OpSystems/.../cpuScheduling.htm
5. https://www.andrew.cmu.edu/course/15-440-sp09/.../ln/lecture3.html

I Year MCA – II Semester

MCA204 Computer Networks

Course Objective: ***The importance of computer networks and the Internet. Recognize the different internetworking devices and their functions. Detail about various protocols that are being used in networking.***

Unit -I:

Objective: Introduce the basic concepts of computer networks and various reference models. Explain how communication works in network and internet.

Introduction :

Uses of computer networks, reference models.

Network Hardware reference model:

Transmission media, Narrowband ISDN, Broad band ISDN, ATM.

The data Link layer:

Design Issues, Error detection and correction, Elementary Data Link Protocols, Sliding window protocols: Data link layer in HDLC, Internet and ATM.

Unit- II:

Objective: Introduce the basic concepts of computer networks and various reference models. Explain how communication works in network and internet.

Channel allocation methods:

TDM, FDM, ALOHA, Carrier sense Multiple access protocols, Collision Free protocols – IEEE standard 802 for LANS – Ethernet, Token Bus, Token ring, Bridges.

Network layer Routing Algorithms:

Shortest path, Flooding, Flow based Distance vector, Link state, Hierarchical, Broadcast routing, Congestion Control algorithms-General principles of congestion control, Congestion prevention polices, Choke packets and Load shedding.

Unit- III

Objective: Recognize the different internetworking devices and their functions. Explain the role of protocols in networking.

Internet Working :

Tunneling, internetworking, Fragmentation, network layer in the internet – IP protocols, IP address, Subnets, Internet control protocols, OSPF, BGP, Internet multicasting, Mobile IP. Network layer in the ATM Networks – cell formats, connection setup, routing and switching, service categories, and quality of service, ATM LANs.

Unit -IV:

Objective: Analyze the services and features of the various layers of data networks

The Transport Layer:

Elements of transport protocols – addressing, establishing a connection, releasing connection, flow control and buffering and crash recovery, end to end protocols : UDP, reliable Byte Stream (TCP) end to end format, segment format, connection establishment and termination, sliding window revisited, adaptive retransmission, TCP extension, Remote Procedure Call – BLAST, CHAN, SELECT, DCE.

Unit- V:

Objective: Analyze the features and operations of various application layer protocols such as HTTP, DNS and SMTP

Application Layer:

Network Security, Cryptographic Algorithms: DES, RSA. Security Mechanisms : Authentication Protocols, Firewalls.Name service (DNS) Domains Hierarchy, Name servers. Traditional Applications : SMTP, MIME, World Wide Web : HTTP, Network Management : SNMP

Text Books:

1. Computer Networks Andrew.S. Tanenbaum, 4/e, Pearson
2. Data and computer communications, stallings, 8/e, PHI

References:

1. Data communications and networking Forouzan, 4/e, TMH
2. Computer Networks – A System Approach , Peterson ,Bruce Davie,2/e,Harcourt Asia
3. Computer communications and networking technologies, Gallo, Hancock,Cengage
4. An Engineering approach to computer networking, Kesha ,Pearson
5. Communication networks, 2/e , Leon-Garcia, TMH
6. Computer networks, Anuranjan Misra, ACME Learning
7. Computer networks, C R Sarma, Jaico,
8. Understanding data communications, Held, 7/e , Pearson

Web Resources:

1 . https://www.ischool.utexas.edu/~i380kpd/reference_models.doc
1. https://en.wikipedia.org/wiki/Networking_hardware
2. https://en.wikipedia.org/wiki/Transport_layer
3. www.tutorialspoint.com/...computer_network/transport_layer_introducti..

I Year MCA – II Semester

MCA205 File Structure

Course Objective: ***Introducing techniques for organization and manipulation of data in secondary storage including the low level aspects of file manipulation which include basic file operations, secondary storage devices and system software.***

Unit –I

Objective:

To introduce the primary design issues that characterize file structure design. Introduce the notions of file structure literacy and conceptual tool kit for file structure design

Introduction:

File Structures: The Heart of the file structure Design, A Short History of File Structure Design, Fundamental File Operations: Physical Files and Logical Files, Opening Files, Closing Files, Reading and Writing, Seeking, Special Characters, The Unix Directory Structure, Physical devices and Logical Files, File-related Header Files, UNIX file System Commands; Secondary Storage and System Software: Disks, Magnetic Tape, Disk versus Tape; CD-ROM: Introduction, Physical Organization, Strengths and Weaknesses; Storage as Hierarchy, A journey of a Byte, Buffer Management, I/O in Unix.

Unit – II

Objective: To introduce file structure concepts dealing with string files and examine issues of portability and standardization

Fundamental File Structure Concepts, Managing Files of Records: Field and Record Organization, Record Access, More about Record Structures, File Access and File Organization.

Organization of Files for Performance, Indexing:

Data Compression, Reclaiming Space in files, Internal Sorting and Binary Searching, Keysorting; What is an Index? A Simple Index for Entry- Sequenced File, Indexes that are too large to hold in Memory, Indexing to provide access by Multiple keys, Retrieval Using Combinations of Secondary Keys, Improving the Secondary Index structure: Inverted Lists, Selective indexes, Binding.

Unit – III

Objective: To describe a class of frequently used processing activities known as consequential processes

Consequential Processing and the Sorting of Large:

FILES: A Model for Implementing Consequential Processes, Extension of the Model to include Multiway Merging, A Second Look at Sorting in Memory, Merging as a Way of Sorting Large Files on Disk.

Multi-Level Indexing and B-Trees:
The invention of B-Tree, Statement of the problem, Indexing with Binary Search Trees; Multi-Level Indexing, B-Trees, Example of Creating a B-Tree, Formal Definition of B-Tree Properties, Worst-case Search Depth, Deletion, Merging and Redistribution, Redistribution during insertion; B* Trees, Buffering of pages; Virtual B-Trees; Variable-length Records and keys.

Unit -IV

Objective: To introduce indexed sequential file and to describe operations on a sequence set of blocks that maintains records in order by key.

Indexed Sequential File Access and Prefix B + Trees:
Indexed Sequential Access, Maintaining a Sequence Set, Adding a Simple Index to the Sequence Set, The Content of the Index: Separators Instead of Keys, The Simple Prefix B+ Tree and its maintenance, Index Set Block Size, Internal Structure of Index Set Blocks: A Variable-order B- Tree, Loading a Simple Prefix B+ Trees, B-Trees, B+ Trees and Simple Prefix B+ Trees in Perspective.

Unit – V

Objective: To introduce the concept of hashing and examine the problem of choosing a good hashing algorithm present a reasonable one in detail and describe some others

Hashing:
Introduction, A Simple Hashing Algorithm, Hashing Functions and Record Distribution, How much Extra Memory should be used?, Collision resolution by progressive overflow, Buckets, Making deletions, Other collision resolution techniques, Patterns of record access
.

Extendible Hashing:
How Extendible Hashing Works, Implementation, Deletion, Extendible Hashing Performance, Alternative Approaches.

Text Book:
1. File Structures-An Object Oriented Approach with C++ -
Michael J. Folk, Bill Zoellick, Greg Riccardi, 3rd Edition, Pearson Education, 1998.
(Chapter 1 to 12 excluding 1.4,1.5,5.5,5.6,8.6,8.7,8.8)

References :

1. File Structures Using C++ - K.R. Venugopal, K.G. Srinivas, P.M.Krishnaraj, Tata McGraw-Hill, 2008.
2. C++ Components and Algorithms - Scot Robert Ladd, BPB Publications, 1993.
3. Database Management Systems - Raghu Ramakrishan and Johannes Gehrke, 3rd Edition, McGraw Hill, 2003.

Web Resources:

1. www.powershow.com/.../Chap1_Introduction_to_**File_Structures**_power...
2. www.ceng.metu.edu.tr/courses/.../week3_Sequential**Files**_section2.pdf
3. https://prezi.com/uge8kluipjco/**managing-files-of-records/**
4. https://www.cse.iitb.ac.in/~sudarsha/db-book/slide-dir/ch12.ppt
5. https://www.site.uottawa.ca/~lucia/courses/2131-02/lect13.pd
6. www.powershow.com/.../**Cosequential_Processing**_powerpoint_ppt_pres...

I Year MCA – II Semester

MCA206 Object Oriented Programming with C++ Lab

Course Objective: ***To introduce the laboratory computing environment , editing, compiling and execution of a program and also simple operation/manipulation commands***

Exercise 1:
Write a C++ program to print the Fibonacci series 0 1 1 2 3 5 8 13 .... By getting number of number to be displayed is given as input

Exercise 2:
Write a C++ program to print the given number in reverse order. Use functions with return type and without return type for reversing the number

Exercise 3:
Create a Structure called employee with the following details as variables within it.
1. Name of the employee
2. Age
3. Designation
4. Salary

Write a C++ program to create array of objects for the structure to access these and print the name, age, designation and salary

Exercise 4:
Create a Union called student with the following details as variables within it.
1. Name of the student
2. Age
3. Year of study
4. Semester
5. 5 different subject marks in array

Write a C++ program to create object for the union to access these and print the Name, age, year, semester and grade according to their percentage of marks scored.
90 % and above – S grade
80% to 89% -- A grade
70% to 79% -- B grade
60% to 69% -- C grade
50% to 59% -- D grade
<50% -- F grade

Exercise 5:
Write a C++ program to print the following by reading number of rows to be printed from the user

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

Exercise 6:
Write a C++ program to find the sum of factorial of a given number using recursive function

Exercise 7:
Write a C++ program to swap two number by both call by value and call by reference mechanism, using two functions swap_value() and swap_reference respectively , by getting the choice from the user and executing the user's choice by switch-case.

Exercise 8:
An electricity board charges the following rates to domestic users ti discourage large consumption of energy:
FOR the first 100 units - 60P per unit
For next 200 units - 80P per unit
Beyond 300 units - 90P per unit
All users are charged a minimum of Rs.50.00. if the total amount is more than Rs.300.00 than an additional surcharge of 15% is added
Write a C++ program to read the names of users and number of units consumed and print out the charges with names

Exercise 9:
An election is contested by five candidates. The candidates are numbered 1 to 5 and a voting is done by marking the candidate number in a ballot paper. Write a C++ program to read the ballot and count the votes cast for each candidate using an array variable count. In case, a number read is outside the range 1 to 5 the ballot should be considered as a 'spoilt ballot', and the program should also count the number of spoilt ballots

Exercise 10:
Create a class for counting the number of objects created and destroyed within various block using constructor and destructors.

Exercise 11:
Write a C++ program to demonstrate the static and non static variable usage defining them within a function

Exercise 12:
Write a C++ program to perform different arithmetic operation such as addition, subtraction, division, modulus and multiplication using inline function

Exercise 13:
Write a C++ program to find the number of vowels present in the given character array using pointer arithmetic.

Exercise 14:
Write a C++ program with different class related through multiple inheritance and demonstrate the use of different access specifiers by means of member variables and member functions.

Exercise 15:
Write a C++ program to explain virtual function (polymorphism) by creating a base class polygon which has virtual function area(). Two classes c_rectangle and c_traingle derived

from c_polygon and they have area() to calculate and return the area of rectangle and triangle respectively.

Exercise 16:
Write a C++ program to create two objects of a class called company and add their data members using an operator overloaded function for '+' operator and '-'operator

Exercise 17:
Write a C++ program to create a class called COMPLEX and implement the following overloading functions ADD that return a COMPLEX number. I. ADD (a, s2) - where a is an integer (real part) and s2 is a complex number. II. ADD (s1, s2)-where s1 & s2 are complex numbers.

Exercise 18:
Write a program to explain class template by creating a template T for a class named pair having two data members of type T which are inputted by a constructor and a member function get-max() return the greatest of two numbers to main. Note: the value of T depends upon the data type specified during object creation

Exercise 19:
Write a C++ program to create three objects for a class named pntr_obj with data members such as roll_no & name . Create a member function set_data() for setting the data values and print() member function to print which object has invoked it using 'this' pointer

Exercise 20:
Write a C++ program to count the number of persons inside a bank, by increasing count whenever a person enters a bank, using an increment(++) operator overloading function, and decrease the count whenever a person leaves the bank using a decrement(--) operator overloading function inside a class

Exercise 21:
Write a program to accept five different numbers by creating a class called friendfunc1 and friendfunc2 taking 2 and 3 arg respectivly and calculate the average of these numbers by passing object of the class to friend function.

Exercise 22:
Write a program to accept the student detail such as name and 3 different marks by get_data() method and display the name and average of marks using display() method. Define a friend class for calculating the average of marks using the method mark_avg().

Exercise 23:
Write a program that creates a binary file by reading the data for the students from the terminal.

Exercise 24:
Write a C++ program which opens a file in reading and writing mode. After writing information inputted by the user to a file, the program reads information from the file and outputs it onto the screen.

I Year MCA – II Semester

MCA207 UNIX Programming Lab

Course Objective: ***Various Unix utilities and shell scripting ,basic understanding of UNIX OS,UNIX commands and File system and to familiarize students with the Linux environment. To make student learn fundamentals of shell scripting and shell programming. Emphases are on making student familiar with UNIX environment and issues related to it.***

1. Programs using basic network commands
2. Program using system calls : create, open, read, write, close, stat, fstat, fseek
3. Program to implement inter process communication using pipes
4. Write a shell script for sorting, searching and insertion/deletion of elements in a list
5. Create two processes to run a for loop, which adds numbers 1 to n, say one process adds odd
   numbers and the other even
6. By creating required number of processors, simulate a communication between them as below:
7. Create a file that is shared among some users, write a program that finds whether a specific user has created read and write operations on the file
8. Create a shared lock and exclusive lock among some number of processes, say 1 to 10 on any
   data of 100 elements. For example, process 5 wants a shared lock on elements 5 to 50 or process 8 wants exclusive lock on elements 32 to 45. Create access violations on the locks and show what occurs, then.
9. Write a program demonstrating semaphore operation on a shared file for reading but not writing
10. Write a program which reads a source file name and destination file name using command line arguments and then converts into specified format (i.e. either from lower case to upper case or upper case to lower case or inverse of each)
11. Write a program which takes a set of filenames along with the command line and print them based on their size in bytes either ascending or descending order
12. Write a program which takes directory name along the command line and displays names of the files which are having more than one link
13. Write a program to demonstrate the use of exec family functions
14. Write a program to display the good morning, good afternoon, good evening and good night
    depending on the users log on time
15. Write a program to demonstrate the working of simple signal handler that catches either of the two user defined signals and prints the signal number
16. Write a program to demonstrate the locking mechanism while accessing the shared files
17. Write a shell script containing a function mycd() using which, it is possible to shuttle between directories
18. Write a shell script which works similar to the wc command. This script can receive the option -l, -w, -c to indicate whether number of lines/words/characters
19. Write a program to print prime numbers between x and y Write a shell script which deletes all lines containing the word

20. Write a shell script which deletes all lines containing the word "UNIX" in the files supplied as arguments to this shell script
21. Write a shell script which displays a list of all files in the current directory to which you have read, write and execute permissions
22. Write a menu-driven program which has the following options:
23. Write a shell script for renaming each file in the directory such that it will have the current shell's PID as an extension. The shell script should ensure that the directories do not get renamed.

II Year MCA - III Semester

MCA301 Java Programming

Course Objective: ***To impart knowledge about java basics and the structure of java program and also able to build dynamic user interface using applets & event handling.***

Unit – I

Objective: To introduce basic concepts of Object Oriented Programming, java data types, classes and objects.

Basics of Object Oriented Programming (OOP):

Need for OO paradigm , A way of viewing world - Agents, responsibility, messages,methods,classes and instances, class hierarchies(Inheritance), method binding, overriding and exceptions, summary of oop concepts, coping with complexity , abstraction mechanisms.

Java Basics:

Data types, variables, scope and life time of variables, arrays, operators, expressions, control statements, type conversion and costing, simple java program, classes and objects - concepts of classes, objects, constructors methods, access control, this keyword, garbage collection, overloading methods and constructors, parameter passing, recursion, string handling.

Unit – II

Objective: To implement object oriented constructs such as various class hierarchies, interfaces & exception handling

Inheritance:

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, abstract classes.

Packages and Interfaces:

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.

Unit – III

Objective: To gain knowledge of throwing on exception and catching it.

Exception handling and multithreading:

Concepts of exception handling, benefits of exception handling, Termination or presumptive models, exception hierarchy, usage of try, catch, throw,throws and finally, built in exceptions, creating own exception sub classes. Differences between multi threading and multitasking, thread life cycle, creating threads, synchronizing threads, daemon threads, thread groups.

Unit – IV

Objective: Being able to build dynamic user interface using event handling and to designing the user interface using the UI components.

Event Handling:

Events, Event sources, Event classes, Event Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes, inner classes. The AWT class hierarchy, user interface components-labels, button, canvas, scrollbars, text components, check box, check box groups, choices, list panes-scroll pane, dialogs, menu bar , graphics, layout manager- layout manager types- boarder, grid, flow, card and grid bag.

Unit – V

Objective: Knows about the creation of applets & using it and application oriented knowledge on swings.

Applets:

Concepts of Applets, differences between applets and applications, lifecycle of an applet, types of applets, creating applets, passing parameters to applets.

Swings:

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 - 7/e, Herbert schildt, TMH.

References:

1. JAVA: How to program, 8/e, Dietal, Dietal, PHI.
2. Introduction of programming with JAVA, S.Dean, TMH.
3. Introduction to Java programming, 6/e, Y.Daniel Liang, Pearson.
4. Core Java 2, Vol 1(Vol 2) Fundamentals (Advanced), 7/e, Cay.S.Horstmann, Gary Cornell, Pearson.
5. Big Java2, 3/e, Cay.S. Horstmann, Wiley.
6. Object Oriented Programming through Java, P.Radha Krishna, University Press.

Web Resources:

1. http://ebook-dl.com/item/java_programming_seventh_edition_joyce_farrell/
2. http://introcs.cs.princeton.edu/java/10elements/
3. http://www.ntu.edu.sg/home/ehchua/programming/java/J3a_OOPBasics.html
4. http://www.horstmann.com/bigjava2.html
5. https://www.edx.org/course/introduction-programming-java-part-1-uc3mx-it-1-1x

II Year MCA - III Semester

MCA302 Database Management Systems

Course Objective: ***To understand the fundamentals of data models and conceptualize, depict a database system using ER diagram, makes a study of SQL, relational database design and write queries using SQL, get knowledge to normalize the data and to know about data storage techniques and query processing.***

Unit – I
Objective: To know about the database system applications, data models, Database design and ER Diagrams.

Data base System Applications, data base 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 applications Programs – data base Users and Administrator – Transaction Management – data base 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
Objective: To introduce relational model and relational algebra.

Relational Model:
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.

Relational Algebra and Calculus:
Relational Algebra – Selection and projection set operations – renaming – Joins – Division – Examples of Algebra overviews – Relational calculus – Tupple relational Calculus – Domain relational calculus – Expressive Power of Algebra and calculus.

Unit – III
Objective: Acquiring the knowledge of query evaluation and designing of database applications using normalization

Form of Basic SQL Query – Examples of Basic SQL Queries – Introduction to Nested Queries – Correlated Nested Queries Set –Comparison Operators – Aggregative Operators – NULL values – Comparison using Null values – Logical connectivity's – AND,OR and NOR – Impact on SQL Constructs – Outer Joins – Disallowing NULL values – Complex Integrity Constraints in SQL Triggers and Active Data bases. Schema refinement – Problems Caused by redundancy – Decompositions – Problem related to decomposition – reasoning about FDS –

FIRST, SECOND, THIRD Normal forms – BCNF – Lossless join Decomposition – Dependency preserving Decomposition – Schema refinement in Data base Design – Multi valued Dependencies – forth Normal Form - fifth Normal Form-Inclusion dependencies.

Unit – IV

Objective: To impart knowledge in transaction processing, concurrency control techniques and recovery procedures

Overview of Transaction Management:

ACID Properties – Transactions and Schedules – Concurrent Execution of transaction –Lock Based Concurrency Control – Performance Locking – Transaction Support in SQL – Introduction to Crash recovery.

Concurrency Control:

Serializability, and recoverability – Introduction to Lock Management – Lock Conversions – Dealing with Dead Locks – Specialized Locking Techniques – Concurrency without Locking.

Crash recovery:

Introduction to ARIES – the Log –Other Recovery related Structures – the Write-Ahead Log Protocol – Check pointing – recovering from a System Crash –Media recovery – Other approaches and Interaction with Concurrency control.

Unit – V

Objective: To gain the knowledge on indexing structures.

Overview of Storage and Indexing:

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 .

Overview of Query Evaluation:

Tree Structured Indexing: Intuitions for tree Indexes – Indexed Sequential Access Methods (ISAM) – B+ Trees: A Dynamic Index Structure. Hash Based Indexing: Static Hashing – Extendable hashing – Linear Hashing – Extendable vs. Linear hashing .Storing data in disks and file .The memory hierarchy – Redundant arrays of independent disks.

Text Books:

1. Data base Management Systems- 3/e, Raghurama Krishnan & Johannes Gehrke- TMH.
2. Data base System Concepts- 6/e, Silberschatz, Korth- TMH

References:

1. Data base Management System, 5/e, Elmasri Navathe, Pearson
2. Introduction to Database Systems, 8/e, C.J.Date, Pearson
3. Data base Systems design, Implementation, and Management, 5/e, Rob, Coronel, Thomson
4. Database Management System, Connolly Begg, Pearson
5. Database Management systems, Farcia-Molina Ullman Widom, Pearson

6. Database Management Systems, Majumdr, Bhattacharyya, TMH.
7. Database System Concepts, Peter ROB, Coronel, Cengage.

Web Resources:

1. https://drive.google.com/file/d/0B-XLkLATpEg9N3F6NHBqWld6czg/edit?pli=1
2. http://www.cse.hcmut.edu.vn/~ttqnguyet/CSDL/EbookDB.pdf
3. https://books.google.co.in/books?id=pk5GAQAAIAAJ&redir_esc=y
4. http://www.britannica.com/technology/database-management-system

II Year MCA - III Semester

MCA303 Intellectual Property Rights & Professional Ethics

Course Objective: ***To appreciate the concept of Intellectual Property and recognize different kinds of Intellectual Property and To appreciate the rationale behind IP and underlying premises, to know the position of IP under the Constitution of India, To understand the concepts of Ethics in work environment. To understand the threats in computing environment and the intricacies of accessibility issues and to ensure safe exits when designing the software projects.***

Unit – I
Objective: To introduce the concepts of intellectual property law basics and various types of trademarks.

Intellectual property law basics:
Types of Intellectual property – agencies responsible for Intellectual property registration.

Introduction to law of trademarks:
Purpose and function, types of trademarks, acquisition of trademark rights, trademark selection and searching, trademark registration process.

Unit – II
Objective: To know the concept of IPR and the relevant Law and its practical application.

Introduction to patent law – patent requirements – ownership-patents application process-patent infringement –patent litigation –International patent law-double patenting-patent searching-invention developers and promoters.

Unit – III
Objective: To know the copyright ownership and their disputes.

Copyright ownership, transfer and duration-right to prepare derivative works- rights of distribution –rights of perform the work publicity copyright formalities and registrations – limitations –copyright disputes and international copyright law.

Unit – IV
Objective: To understand the laws for prosecuting computer attacks and risk assessment.

Computer and Internet Crime:
Types of exploits, types of perpetrators, federal laws for prosecuting computer attacks, implementing trustworthy computing, Risk assessment, establishing security policy, educating employees and contract workers, prevention, detection, response.

Unit – V

Objective: To know about the importance of software quality, social networking and ethical issues.

Software Development:

Importance of software quality, software product liability, software development process, capability maturity model integration, key issues in software development, development of safety critical system, quality management standards.

Social Networking –Ethical issues:

Cyber bullying- cyber stalking –online virtual world- crime in virtual world

Text Books:

1. Intellectual Property- Deborah E.Bouchoux- Cengage learning, New Delhi.
2. Ethics in information technology - George W.Reynolds-4/e

References:

1. Prabuddha Ganguli: Intellectual Property Rights Unleashing the Knowledge Economy. Tata McGraw Hill, New Delhi, 2001.
2. Penny Duquenoy, Simon Jones and Barry G Blundell, "Ethical, legal and professional issues in computing", Middlesex University Press, 2008.
3. Jayasree Watal-Intellectual Property Rights: In the WTO and Developing Countries -Oxford University Press.
4. V.Sarkar-Intellectual Property Rights and Copyright- ESS publications.

Web Resources:

1. http://www.cengagebrain.com.au/content/9781285243863.pdf
2. http://www.abebooks.co.uk/book-search/title/intellectual-property-for-paralegals-the-law-of-trademarks-copyrights-patents-and-trade-secrets/author/bouchoux-deborah-e/
3. http://capitadiscovery.co.uk/brookes/items/1194118
4. http://trove.nla.gov.au/work/34006945?q&versionId=41863400

II Year MCA - III Semester

MCA304 Computer Graphics & Vision

Course Objective: ***Introduces the basic concepts of Computer Graphics which shall be useful for virtual modeling. To understand computational development of graphics with mathematics and provide in-depth knowledge of display systems, image synthesis, shape modeling of 2 D & 3D application.***

Unit – I
Objective: To understand contemporary graphics principles and graphics hardware

Introduction:
Application areas of Computer Graphics, overview of graphics systems, video-display devices, raster scan systems, random scan systems, graphics monitors and work stations and input devices.

Output primitives:
Points and lines, line drawing algorithms, mid-point circle and ellipse algorithms. Filled area primitives: Scan line polygon fill algorithm, boundary-fill and flood-fill algorithms.

Unit – II
Objective: Understand and demonstrate geometrical transformations.

2-D geometrical transforms:
Translation, scaling, rotation, reflection and shear transformations, matrix representations and homogeneous coordinates, composite transforms transformations between coordinate systems.

Unit – III
Objective: Understand and demonstrate 2Dobject representation techniques.

2-D viewing:
The viewing pipeline, viewing coordinate reference frame, window to view-port coordinate transformation, viewing functions, Cohen-Sutherland and Cyrus-beck line clipping algorithms, Sutherland –Hodgeman polygon clipping algorithm

Unit – IV
Objective: Understand and demonstrate 3D object representation techniques.

3-D object representation:
Polygon surfaces, quadric surfaces, spline representation, Hermite curve, Bezier curve and B-Spline curves, Bezier and B-Spline surfaces. Basic illumination models, polygon rendering methods.

3-D Geometric transformations:
Translation, rotation, scaling, reflection and shear transformations, composite transformations.

3-D Viewing:
Viewing pipeline, viewing coordinates, view volume and general projection transforms and clipping

Unit – V
Objective: To introduce basic concepts in animation and motion specifications.

Visible surface detection methods:
Classification, back-face detection, depth-buffer, scan-line, depth sorting, BSP tree methods, area sub-division and octree methods.

Computer animation:
Design of animation sequence, general computer animation functions, raster animation, computer animation languages, key frame systems, motion specifications.

Text Books:
1. Computer Graphics *C version* - Donald Hearn&M.Pauline Baker- Pearson

References:

1. Computer Graphics- Donald Hearn & M.Pauline Baker- 2/E, PHI
2. Computer Graphics, Zhigand xiang, Roy Plastock, Schaum's outlines, 2/E, TMH
3. Computer Graphics Principles & practice, 2/e, Foley, VanDam, Feiner, Hughes, Pearson
4. Procedural elements for Computer Graphics, David F Rogers, 2/e, TMH
5. Principles of Interactive Computer Graphics, Neuman, Sproul, TMH.
6. Principles of Computer Graphics, Shalini Govil, Pai, 2005, Springer.
7. Computer Graphics, Steven Harrington, TMH

Web Resources:
1. http://www.ddegjust.ac.in/studymaterial/mca-3/ms-13.pdf
2. http://www.gehu.ac.in/uploads/pdf/lab.pdf
3. http://www.uptu.ac.in/pdf/sub_ecs_504_30sep14.pdf
4. http://www.pearsoned.co.in/web/books/9788131705056_Computer-Graphics_James-D-Foley.aspx

II Year MCA - III Semester

MCA305 Design & Analysis of Algorithms

Course Objectives: ***To study paradigms and approaches used to analyze and design algorithms and to appreciate the impact of algorithm design in practice. It also ensures that students understand how the worst-case time complexity of an algorithm is defined, how asymptotic notation is used to provide a rough classification of algorithms***

Unit – I

Objective: It also ensures that students understand how the worst-case time complexity of an algorithm is defined, how asymptotic notation is used to provide a rough classification of algorithms.

Basic Concepts of Algorithms:

Introduction - Notion of Algorithm - Fundamentals of Algorithmic Solving - Important Problem types - Fundamentals of the Analysis Framework - Asymptotic Notations and Basic Efficiency Classes.

Mathematical Aspects and Analysis of Algorithms:

Mathematical Analysis of Non-recursive Algorithm - Mathematical Analysis of Recursive Algorithm. (Performance Analysis- Space Complexity and Time Complexity. Disjoint Set, Disjoint Set Operation, Union and find algorithm).

Unit – II

Objective: To study paradigms and approaches used to analyze sorting algorithms and to appreciate the impact of algorithm design in practice.

Analysis of Sorting and Searching Algorithms:

Brute Force - Selection Sort And Bubble Sort - Sequential Search And Brute-Force String Matching - Divide And Conquer - Merge Sort - Quick Sort - Binary Search - Binary Tree-Traversal And Related Properties - Decrease And Conquer - Insertion Sort - Depth First Search And Breadth First Search.

Unit – III

Objective: Synthesize dynamic-programming algorithms, and analyze them.

Algorithmic Techniques:

Transform and conquer - Presorting - Balanced Search trees - AVL Trees - Heaps and Heap sort - Dynamic Programming – Matrix Chain Multiplication ,0/1 Knapsack Problem, Travelling Salesman Problem,Warshall's and Floyd's Algorithm - Optimal Binary Search trees - Greedy Techniques - Prim's Algorithm - Kruskal's Algorithm - Dijkstra's Algorithm - Huffman trees.

Unit – IV

Objective: To introduce the backtracking and branch and bound algorithms.

Algorithm Design Methods -Backtracking - n-Queen's Problem - Hamiltonian Circuit problem - Subset-Sum problem - Branch and bound - Assignment problem - Knapsack problem - Traveling salesman problem.

Unit – V

Objective: Know the concepts of tractable and intractable problems and the classes P, NP and NP-complete problems.

Branch and Bound:

LC Branch and Bound solution, FIFO Branch and Bound solution.

NP-Hard and NP-Complete problems:

Basic concepts, non deterministic algorithms, NP - Hard and NP Complete classes, Cook's theorem.

Text Books:

1. The Design and Analysis of Computer Algorithms - A.V.Aho, J.E. Hop croft and J.D.Ullman- Pearson Education Asia, 2003.
2. Introduction to Algorithms -T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein- PHI Pvt. Ltd., 2001

References:

1. Computer Algorithms - Introduction to Design and Analysis- Sara Baase and Allen Van Gelder, Pearson Education Asia, 2003.
2. Introduction to the Design and Analysis of Algorithm- Anany Levitin, Pearson Education Asia, 2003.

Web Resources:

1. https://books.google.co.in/books?id=FKcCHr9NxjYC&pg=PA47&lpg=PA47&dq=1.%09The+Design+and+Analy
2. ftp://doc.nit.ac.ir/cee/jazayeri/Algorithms/Books/Design%20&%20Analysis%20of%20Algorithm.pdf
3. https://books.google.co.in/books/about/Design_Analysis_Of_Algorithms.html?id=FKcCHr9NxjYC
4. http://cc.ee.ntu.edu.tw/~ywchang/Courses/PD/EDA_Chapter4.pdf

II Year MCA - III Semester

MCA306 Java Programming Lab

Course Objectives: ***Understand fundamentals of programming such as variables, conditional and iterative execution, methods, etc.one can able to define classes, invoking methods, using class libraries, etc.Attain the ability to write a computer program to solve specified problems. Be able to use applets and swings to create and run simple Java programs.***

Note:

- Use JDK 1.5 or above on any platform e.g. Windows or UNIX.
- Student needs exposure to all programs, but expected to complete atleast 15.

1. The Fibonacci sequence is defined by the following rule. The first 2 values in the sequence are 1, 1. Every Subsequent value is the sum of the 2 values preceding it. Write a Java Program (WAJP) that uses both recursive and non-recursive functions to print the *nth* value of the Fibonacci sequence.
2. WAJP to demonstrate wrapper classes, and to fix the precision.
3. WAJP that prompts the user for an integer and then prints out all the prime numbers up to that Integer.
4. WAJP that checks whether a given string is a palindrome or not. Ex: MALAYALAM is a palindrome.
5. WAJP for sorting a given list of names in ascending order.
6. WAJP to check the compatibility for multiplication, if compatible multiply two matrices and find its transpose.
7. WAJP that illustrates how runtime polymorphism is achieved.
8. WAJP to create and demonstrate packages.
9. WAJP, using String Tokenizer class, which reads a line of integers and then displays each integer and the sum of all integers.
10. WAJP that reads on file name from the user then displays information about whether the file exists, whether the file is readable/writable, the type of file and the length of the file in bytes and display the content of the using File Input Stream class.
11. WAJP that displays the number of characters, lines and words in a text/text file.
12. Write an Applet that displays the content of a file.
13. Write a program using applets for passing parameters.
14. WAJP that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the + - x / %operations. Add a text field to display the result.
15. WAJP for handling mouse events.
16. WAJP demonstrating the life cycle of a thread.

17. WAJP that correctly implements Producer-Consumer problem using the concept of Inter Thread Communication.
18. WAJP that lets users create Pie charts. Design your own user interface (with Swings & AWT).
19. WAJP that allows user to draw lines, rectangles and ovals.
20. WAJP to generate a set of random numbers between two numbers x1 and x2, and x1>0.
21. WAJP to create an abstract class named Shape, that contains an empty method named number Of Sides ().Provide three classes named Trapezoid, Triangle and Hexagon, such that each one of the classes contains only the method number Of Sides (), that contains the number of sides in the given geometrical figure.
22. WAJP to implement a Queue, using user defined Exception Handling (also make use of throw, throws).
23. WAJP that creates 3 threads by extending Thread class. First thread displays "Good Morning" every 1 sec, the second thread displays "Hello" every 2 seconds and the third displays "Welcome" every 3 seconds. (Repeat the same by implementing Runnable)
24. Create an inheritance hierarchy of Rodent, Mouse, Gerbil, Hamster etc. In the base class provide methods that are common to all Rodents and override these in the derived classes to perform different behaviors, depending on the specific type of Rodent. Create an array of Rodent, fill it with different specific types of Rodents and call your base class methods.
25. WAJP to create and add 3 scroll bars to the window and handle the event appropriately.
26. WAJP on key event handling.
27. WAJP which uses flow layout.
28. WAJP which uses grid layout.

II Year MCA - III Semester

MCA307 Database Management Systems Lab

Course Objectives: ***To familiarize the participant with the nuances of database environments and give a good formal foundation on the relational model of data. To present SQL and procedural interfaces to SQL comprehensively. To give an introduction to systematic database design approaches covering conceptual design, logical design and an overview of physical design. To present the concepts and techniques relating to query processing by SQL engines. To present the concepts and techniques relating to ODBC and its implementations. To introduce the concepts of transactions and transaction processing. To present the issues and techniques relating to concurrency and recovery in multi-user database environments***

SQL

1. Consider the insurance database given below. The primary keys are made bold and the data types are specified.
PERSON(**driver_id**:string , name: string , address: string)
CAR(**regno**:string , model: string , year:int)
ACCIDENT(**report_number**:int , accd_date:date , location: string)
OWNS(**driver_id**:string , **regno**:string)
PARTICIPATED(**driver_id**:string , **regno**:string , **report_number**:int , damage_amount:int)
1)Create the above tables by properly specifying the primary keys and foreign keys.
2) Enter at least five tuples for each relation.
3) Demonstrate how you (a) update the damage amount for the car with specific regno in the accident with report number 12 to 25000. (b) Add a new accident to the database.
4) Find the total number of people who owned cars that were involved in accidents in the year 2008.
5) Find the number of accidents in which cars belonging to a specific model were involved.

2. Consider the following relations for a order processing database application in a company.
CUSTOMER(**custno**:int , cname:string , city: string)
ORDER(**orderno**:int , odate:date , custno:int , ord_amt:int)
ORDER_ITEM(**orderno**:int , **itemno**:int , quantity:int)
ITEM(**itemno**:int , unitprice:int)
SHIPMENT(**orderno**:int , **warehouseno**:int , ship_date:date)
WAREHOUSE(**warehouseno**:int , city:string)
1)Create the above tables by properly specifying the primary keys and foreign keys.
2) Enter at least five tuples for each relation.
3) Produce a listing: custname, No_of_orders, Avg_order_amount, where the middle column is the total number of orders by the customer and the last column is the average order amount for that customer.

4) List the orderno for orders that were shipped from all the warehouses that the company has in a specific city.
5) Demonstrate the deletion of an item from the ITEM table and demonstrate a method of handling the rows in the ORDER_ITEM table that contains this particular item.

3. Consider the following database of student enrollment in courses and books adopted for that course.
STUDENT(**regno**:string , name: string , major: string , bdate:date)
COURSE(**courseno**:int , cname:string , dept:string)
ENROLL(**regno**:string , **courseno**:int , **sem**:int , marks:int)
BOOK_ADOPTION(**courseno**:int , **sem**:int , book_isbn:int)
TEXT(**book_isbn**:int , book_title:string , publisher: string , author: string)
1)Create the above tables by properly specifying the primary keys and foreign keys.
2) Enter atleast five tuples for each relation.
3) Demonstrate how you add a new text book to the database and make this book to be adopted by some department.
4) Produce a list of text books (includes courseno, book_isbn, book_title) in the alphabetical order for courses offered by the 'CS' department that use more than two books.
5) List any department that has all its books published by a specific publisher.

4. The following are maintained by a book dealer.
AUTHOR(**author_id**:int , name: string , city: string , country: string)
PUBLISHER(**publisher_id**:int , name: string , city: string , country: string)
CATALOG(**book_id**:int , title: string , author_id:int , publisher_id:int , category_id:int , year:int , price:int)
CATEGORY(**category_id**:int , description: string)
ORDER_DETAILS(**order_no**:int , **book_id**:int , quantity:int)
1)Create the above tables by properly specifying the primary keys and foreign keys.
2) Enter at least five tuples for each relation.
3) Give the details of the authors who have 2 or more books in the catalog and the price of the books is greater than the average price of the books in the catalog and the year of publication is after 2000.
4) Find the author of the book that has maximum sales.
5) Demonstrate how you increase the price of books published by a specific publisher by 10%.

5. Consider the following database for a banking enterprise.
BRANCH(**branch_name**:string , branch_city:string , assets: real)
ACCOUNT(**accno**:int , branch_name:string , balance: real)
DEPOSITOR(**customer_name**:string , **accno**:int)
CUSTOMER(**customer_name**:string , customer_street:string , customer_city:string)
LOAN(**loan_number**:int , branch_name:string , amount: real)

BORROWER(**customer_name**:string , **loan_number**:int)

1)Create the above tables by properly specifying the primary keys and foreign keys.

2) Enter at least five tuples for each relation.

3) Find all the customers who have at least two accounts at the main branch.

4) Find all the customers who have an account at all the branches located in a specific city.

5) Demonstrate how you delete all account tuples at every branch located in a specific city.

PL/SQL

6. Write a PL/SQL code block that will accept an account number from the user and debit an amount of Rs. 2000 from the account if the account has a minimum balance of 500 after the amount is debited. The Process is to fired on the Accounts table.
7. Write a PL/SQL code block to calculate the area of the circle for a value of radius varying from 3 to 7. Store the radius and the corresponding values of calculated area in a table Areas.
8. Write a PL/SQL block of code for inverting a number 5639 or 9365.
9. Write a PL/SQL block of code to achieve the following: if the price of Product 'p00001' is less than 4000, then change the price to 4000. The Price change s to be recorded in the old_price_table along with Product no and the date on which the price was last changed. Tables involved: product master- product no, sell price.
10. Write a Pl/Sql block which displays area for a given radius. If no data found then display an error message.
11. Write a PL/SQL block that will display the name, department and salary of the first 10 employees getting the highest salary.
12. A HRD manager has decided to raise the salary for all the employees in Department number 20 by 0.05. Whenever any such raise is given to the Employees, an audit trail of the same is maintained in the emp_raise table. The emp_raise table holds the employee number the date when the raise was given and the raise amount. Write a PL/SQL block to update the Salary of each employee of dept_no 20 appropriately and insert a record in the emp_raise table as well.

    Employee- Emp_code, Ename, Dept_no, job, salary.

    Emp_raise-Emp_code, Raise_date, Raise_amt.
13. Repeat the above problem using cursor for loops.
14. Write a row trigger to insert the existing values of the salary table into a new table when the salary table is updated
15. Write a PL/SQL block of code that lists the highest salary drawn by an employee in each of the departments. It should make use of a function dept_highest which return the highest salary drawn by an employee for the given department.
16. Create a package to include the following:

    A named procedure to list the employee names of given department

    A function which returns the max salary for the given department

II Year MCA – IV Semester

MCA401 Web Technologies

Course Objective: ***To impart knowledge about various technologies that can be used in developing web pages like JavaScript ,HTML, XML , Server side programming with Java Servelets and JSP, database connectivity using JDBC***

Unit –I
Objective: To introduce the concepts of HTML and JavaScript and its usage in developing Web Pages.

Review of HTML4
Common tags, HTML Tables and formatting internal linking, Complex HTML forms.Introduction to Scripting Languages: Java Scripts, Control structures, functions, arrays & objects, DHTML, CSS, event model, filters & transitions.[Text Book 1]

Unit - II
Objective: To impart knowledge of XML and its usage in storing data for Web Programming.

XML
Working with XML, Document type definition, XML Schemas, Document Object Model, XSLT, DOM and SAX. [Text Book 1]

Unit - III
Objective: To introduce the concepts of Java Beans and Servlets .Use of Java Beans in storing data and Servelets for Server side Programming.

Java Beans
Introduction to Java Beans, Advantages of Java Beans, BDK, Introspection, Using Bound properties, Bean Info Interface, Constrained properties, Persistence, Customizers, Java Beans API. [Text Book 2]

Introduction to Servelets
Lifecycle of a Servelet, JSDK, The Servelet API, The javax.servelet Package, Reading Servelet parameters, Reading Initialization Parameters, The javax.servelet.HTTP package, Handling, Http Request & responses, Using Cookies, Session Tracking, Security Issues. [Text Book 2]

Unit - IV
Objective: To introduce the concepts of JSP,Installation of JSDK and use of JSP for developing Server Pages.

Introduction to JSP
The Problem with Servelets, The Anatomy of a JSP Page, JSP Processing, JSP Application Design with MVC,Setting up the JSP Environment: Installing the Java Software Development Kit, Tomcat Server & Testing Tomcat. [Text Book 2]

JSP Application Development

Generating Dynamic Content, Using Scripting Elements, Implicit JSP Objects, Conditional Processing – Displaying Values, Using an Expression to Set an Attribute, Declaring Variables and Methods, Error Handling and Debugging, Sharing Data Between JSP Pages, Requests, and Users, Passing Control and Data Between Pages – Sharing Session and Application Data Memory Usage Considerations. [Text Books 2 & 3]

Unit -V

Objective: To impart knowledge of JDBC and database programming .Use of different drivers for connecting with different databases.

Database Access: Database Programming using JDBC, Studying Javax.sql.* package. Accessing a Database from a JSP Page, Application – Specific Database Actions Deploying JAVA Beans in a JSP Page.

Text Books:

1. Internet and World Wide Web: How to program, 6/e, Dietel, Dietel, Pearson.
2. The Complete Reference Java2, 3/e, Patrick Naughton, Herbert Schildt, TMH.
3. Java Server Faces, Hans Bergstan, O'reilly.

References:

1. Web Programming, building internet applications, 2/e, Chris Bates, Wiley Dreamtech
2. Programming World Wide Web, Sebesta, PEA
3. Web Tehnologies, 2/e, Godbole, kahate, TMH
4. An Introduction to web Design, Programming, Wang, Thomson.
5. Web Technologies, 1/e,A.A.Putambekar, Technical publications.

Web Resources:

1. www.sssi.org.au/userfiles/event_doc1320138204.pdf
2. http://catalogue.pearsoned.co.uk/samplechapter/0130428205.pdf
3. http://www.tutorialspoint.com/web_developers_guide/web_basic_concepts.htm

II Year MCA – IV Semester

MCA402 Software Engineering

Course Objective: ***To impart knowledge of software process models , software requirements and SRS document, software architectural styles,software testing , quality control and how to ensure good quality software.***

Unit- I
Objective: To understand the basic features of software and how to change in nature.

Introduction to Software Engineering:
The evolving role of software, Changing Nature of Software, Software myths.(Text Book 2)

The software problem:
Cost, schedule and quality, Scale and change. (Text Book 1)

Unit- II
Objective: To introduce the software process models such as waterfall and evolutionary models.

Software Process:
Process and project, component software process, Software development process models : Waterfall model, prototyping, iterative development, relational unified process, time boxing model, Extreme programming and agile process, using process models in a project. Project management process.(Text Book 1)

Unit- III
Objective: To introduce the software requirements and SRS document.

Software requirement analysis and specification:
Value of good SRS, requirement process, requirement specification, functional specifications with use-cases, other approaches for analysis, validation.

Planning a software project:
Effort estimation, project schedule and staffing, quality planning, risk management planning, project monitoring plan, detailed scheduling.(Text Book 1)

Unit- IV
Objective: To introduce the different software architectural styles.

Software Architecture:
Role of software architecture, architecture views, components and connector view, architecture styles for C & C view, documenting architecture design, evaluating architectures.

Design:
Design concepts, function-oriented design, object oriented design, detailed design, verification, and metrics. (Text Book 1)

Unit - V

Objective: To provide the basic knowledge in software testing approaches such as unit testing and integration testing and introduce the metrics of the model.

Coding and Unit testing

Programming principles and guidelines, incrementally developing code, managing evolving code, unit testing, code inspection, metrics.(Text Book 1)

Quality Management

Quality concepts, Software quality assurance, Software Reviews, Formal technical reviews, Statistical Software quality Assurance, Software reliability, The ISO 9000 quality standards.(Text Book 2)

Testing

Testing concepts, testing process, black-box testing, white-box testing, metrics.(Text Book 1)

Text Books:

1. Software Engineering, A Precise approach, Pankaj Jalote, Wiley
2. Software Engineering, 3/e & 7e Roger S.Pressman, TMH

References:

1. Software Engineering, 8/e, Sommerville, Pearson.
2. Software Engineering principles and practice, W S Jawadekar, TMH

Web Resources:

1. http://ce.sharif.edu/courses/8485/1/ce474/resources/root/Pressman_Sofware%20Engineering.pdf
2. http://www.slideshare.net/rhspcte/software-engineering-ebook-roger-s-pressman

II Year MCA – IV Semester

MCA403 Data Warehousing and Data Mining

Course Objective: ***To impart knowledge on data mining principles techniques and demonstrate basic data mining algorithms, methods, tools and identify business applications of data mining applying critical thinking, problem-solving, and decision-making skills.***

Unit - I

Objectives: To introduce the data mining principles and data preprocessing.

Introduction

Fundamentals of data mining, Data Mining Functionalities, Major issues in Data Mining.Data Objects and Attribute Types, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity and Dissimilarity.

Data Preprocessing

Needs Preprocessing the Data, Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation.

Unit - II

Objectives: To implement the data warehouse and the development process.

Data Warehouse and OLAP

Data Warehouse and OLAP Technology for Data Mining Data Warehouse, Multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Further Development of Data Cube Technology, From Data Warehousing to Data Mining.

Unit - III

Objectives: To provide the mining patterns, associations and advanced pattern mining.

Mining Frequent Patterns, Associations, and Correlations:

Basic Concepts and Methods. Frequent Item set Mining Methods, Pattern Evaluation Methods.

Advanced Pattern Mining

Pattern Mining: A Road Map, Pattern Mining in Multilevel, Multidimensional Space, Constraint-Based Frequent Pattern Mining, Mining High-Dimensional Data and Colossal Patterns, Mining Compressed or Approximate Patterns, Pattern Exploration and Application.

Unit - IV

Objectives: To introduce the basic concepts ofclassifications and their advanced methods.

Classification:

Basic Concepts, Decision Tree Induction, Bayes Classification methods, Rule-Based Classification,

Advanced Methods:

Bayesian Belief Networks, Classification by Back propagation, Support Vector Machines, Classification Using Frequent Patterns, lazy learners (or learning from your

neighbours),Other Classification Methods.

Unit - V
Objectives: To providing the specifications of advanced clusters analysis and outlier detection.

Cluster Analysis:
Basic Concepts and Methods: Cluster Analysis , Partitioning Methods,
Hierarchical Methods Density-Based Methods,Grid-Based Methods, Evaluation of Clustering

AdvancedClusterAnalysis:
ProbabilisticModel-Based Clustering ,Clustering High-Dimensional Data , Clustering Graph and Network Data Clustering with Constraints.

Text Books:

1. Data Mining Concepts & Techniques "Jiawei Han, Micheline Kamber,Jian Pei,3e.

References:

1. Introduction to data mining: Pang- Ning Tan, Michel Steinbach, Vipin Kumar, Addision –Wesley.
2. Data Mining Concepts & Techniques "Jiawei Han, Micheline Kamber,Jian Pei,1e.
3. Data Mining Techniques, Arun K Pujari, University Press.
4. Data Warehousing Fundamentals, Paulraj Ponnaiah, Wiley.
5. The Data Warehouse Life cycle Tool kit, Ralph Kimball, Wiley .

Web Resources:

1. www.dataminingbook.info/uploads/book.pdf
2. www.cs.rpi.edu/~zaki/PaperDir/DMABOOK.pdf
3. www.ccs1.hnue.edu.vn/hungtd/DM2012/DataMining_BOOK.pdf
4. www.web.engr.illinois.edu/~hanj/bk2/toc.pdf

MCA404 Human Computer Interaction

Course Objective: ***To develop an understanding of user interface design in general, to bring familiarity with the vocabulary associated with sensory and cognitive systems and to make familiar with a variety of both conventional and non-traditional user interface paradigms***

Unit - I
Objective: To develop and understand the user interface designed and importance of designing in general.

Introduction:
Importance of user Interface, definition, importance of good design. Benefits of good design. A brief history of Screen design.

The graphical user interface:
Popularity of graphics, the concept of direct manipulation, graphical system, Characteristics, Web user –interface popularity, characteristics- Principles of user interface.

Unit - II
Objective: To introducetheAlternatives of spatial keyboard and mouse computing.

Design process
Human interaction with computers, importance of human characteristics human consideration, Human interaction speeds, understanding business functions.

Unit - III
Objective: To provide the designing the screen for planning purpose.

Screen Designing
Design goals, Screen planning and purpose, organizing screen elements, ordering of screen data and content, screen navigation and flow, Visually pleasing composition, amount of information, focus and emphasis, presentation information simply and meaningfully, information retrieval on web, statistical graphics, Technological consideration in interface design,system menus and navigation schemes.

Unit - IV
Objective: To introduce the basic navigation tools and components in windows and to use the components.

Windows
Selection of window, selection of Device- Based and Screen-Based controls.

Components
Clear text and messages , Icons and images,choose the proper colors .

Unit - V
Objective: To provide the different software tools and interaction devices in user interface.

Software tools
Specification methods, interface, Building Tools.

Interaction Devices
Keyboard and function keys, pointing devices, speech recognition digitization and generation, image and video displays, drivers.

Text Books:

1. Human Computer Interaction. 3/e, Alan Dix, Janet Finlay, Goryd, Abowd, Russell Beal, PEA, 2004.
2. The Essential guide to user interface design,2/e, Wilbert O Galitz, Wiley DreamTech.

References:

1. Designing the user interface. 4/e, Ben Shneidermann, PEA.
2. User Interface Design, Soren Lauesen, PEA.
3. Interaction Design PRECE, ROGERS, SHARPS, Wiley.
4. Human Computer, Interaction Dan R.Olsan, Cengage, 2010.

Web Resources:

1. http://www.sigchi.org/chi97/proceedings/tutorial/rjkj.htm
2. http://www.idemployee.id.tue.nl/g.w.m.rauterberg/publications/HCI-tutorial.pdf

II Year MCA – IV Semester

MCA405 Cloud Computing

Course Objective: ***To impart fundamental concepts in the area of cloud computing and provide knowledge about applications of cloud computing***

Unit - I
Objective: To impart knowledge about clustering, virtualization and cloud,Performance,Security and other advantages.

Systems modeling, Clustering and virtualization
Scalable Computing over the Internet, Technologies for Network based systems, System models for Distributed and Cloud Computing, Software environments for distributed systems and clouds, Performance, Security and Energy Efficiency

Unit - II
Objective: To introduce the concepts of Virtual machines and virtualization.Its role in Data Center Automation.

Virtual Machines and Virtualization of Clusters and Data Centers
Implementation Levels of Virtualization, Virtualization Structures/ Tools and mechanisms, Virtualization of CPU, Memory and I/O Devices, Virtual Clusters and Resource Management, Virtualization for Data Center Automation.

Unit - III
Objective: To edify the basics of Cloud Platform,Models, Security and SOA.

Cloud Platform Architecture
Cloud Computing and service Models, Architectural Design of Computer and Storage Clouds, Public Cloud Platforms, Inter Cloud Resource Management, Cloud Security and Trust Management. Service Oriented Architecture, Message Oriented Middleware.

Unit - IV
Objective: To impart knowledge about different Cloud Platforms.

Cloud Programming and Software Environments
Features of Cloud and Grid Platforms, Parallel & Distributed Programming Paradigms, Programming Support of Google App Engine, Programming on Amazon AWS and Microsoft Azure, Emerging Cloud Software Environments.

Unit - V
Objective: To introduce the mechanism of resource management and scheduling in Cloud.

Cloud Resource Management and Scheduling
Policies and Mechanisms for Resource Management Applications of Control Theory to Task Scheduling on a Cloud, Stability of a Two Level Resource Allocation Architecture, Feedback Control Based on Dynamic Thresholds. Coordination of Specialized Autonomic Performance Managers, Resource Bundling, Scheduling Algorithms for Computing Clouds, Fair Queuing, Start Time Fair Queuing, Borrowed Virtual Time, Cloud Scheduling Subject to Deadlines,

Scheduling Map Reduce Applications Subject to Deadlines.

Text Book:

1. Distributed and Cloud Computing, Kai Hwang, Geoffrey C. Fox, Jack J. Dongarra MK Elsevier.

References:

1. Cloud Computing, A Practical Approach, Anthony T Velte, Toby J Velte, Robert Elsenpeter, TMH
2. Mastering Cloud Computing, Foundations and Application Programming, Raj Kumar Buyya, Christen vecctiola, S Tammarai selvi, TMH
3. Cloud Computing, Theory and Practice, Dan C Marinescu, MK Elsevier.
4. Cloud Computing, A Hands on approach, Arshadeep Bahga, Vijay Madisetti, University Press.

Web Resources:

1. www.cloudipedia.com/files/2009/11/cloud_computing_made_easy.pdf
2. https://studytm.files.wordpress.com/2014/03/cloud-computing-bible.pdf

II Year MCA – IV Semester

MCA406 ERP& Supply Chain Management

Course Objective: ***To provide the pragmatic knowledge of ERP and supply chain management and how a business process often spans different functional areas of the business: accounting, marketing, material management, etc.***

Unit - I

Objective: To bring awareness about ERP and its benefits,basics of ERP implementation.

Introduction to ERP

Overview – Benefits of ERP, ERP and Related Technologies, Business Process Reengineering, Data Warehousing, Data Mining – Online Analytical Processing, Supply Chain Management.

ERP Implementation

Implementation Life Cycle, Implementation Methodology, Hidden Costs, Organizing Implementation, Vendors, Consultants and Users, Contracts, Project Management and Monitoring.

Unit - II

Objective: To introduce the concepts of business modules and the fundamentals of SCM.

Business Modules

Business Modules in an ERP Package, Finance, Manufacturing, Human Resource, Plant Maintenance, Materials Management, Quality Management, Sales and Distribution.

Fundamentals of Supply Chain Management

Supply chain networks, integrated supply chain planning, Decision phases in supply chain, process view of a supply chain, supply chain flows, Overview of supply chain models and modelling systems, Supply chain planning: Strategic, operational and tactical, Understanding supply chain through process mapping and process flow chart.

Unit - III

Objective: To impart knowledge about SCM strategies, performance,planning and other related concepts.

SCM Strategies, Performance

Supply chain strategies, achieving strategic fit, value chain, Supply chain drivers and obstacles, Strategic Alliances and Outsourcing, purchasing aspects of supply chain, Supply chain performance measurement: The balanced score card approach, Performance Metrics. Planning demand and supply: Demand forecasting in supply chain, Aggregate planning in supply chain, Predictable variability.

Unit - IV

Objective: To introduce the concept of Planning and Managing Inventories and concepts related to Distribution Management.

Planning and Managing Inventories
Introduction to Supply Chain Inventory Management. Inventory theory models: Economic Order Quantity Models, Reorder Point Models and Multiechelon Inventory Systems, Relevant deterministic and stochastic inventory models and Vendor managed inventory models.

Distribution Management
Role of transportation in a supply chain - direct shipment, warehousing, cross-docking; push vs. pull systems; transportation decisions (mode selection, fleet size), market channel structure, vehicle routing problem. Facilities decisions in a supply chain. Mathematical foundations of distribution management, Supply chain facility layout and capacity planning.

Unit - V
Objective: To provide insight of strategic cost management in SCM.

Strategic Cost Management in Supply Chain
The financial impacts, Volume leveraging and cross docking, global logistics and material positioning, global supplier development, target pricing, cost management enablers, Measuring service levels in supply chains, Customer Satisfaction/Value/Profitability/Differential Advantage.

Text Books:

1. ERP Demystified, 2/e, Alexis Leon, TMH, 2007.
2. Supply Chain Management: Strategy, Planning, Operation, Sunil Chopra, Peter Meindel, PEA, 2002.

References:

1. Enterprise Resource Planning, Concepts and Planning, Vinod Kumar, Venkata Krishnan, PHI.
2. Enterprise Resource Planning Concepts and Practice, 7/e, Vinod Kumar, PHI.
3. Supply Chain Management and Advanced Planning, 2/e, Stadtler,Kilger, Springer.

Web Resources:

1. http://www.exforsys.com/tutorials/erp.html

MCA407 Information Storage and Management

Course Objective: ***To impart knowledge on data mining principles techniques and demonstrate basic data mining algorithms, methods, tools and identify business applications of data mining applying critical thinking, problem-solving, and decision-making skills.***

Unit - I

Objective: To impart knowledge about different Business storage, backup, archives and EMC.

IntroductiontoBusinessContinuity

Information Availability, BC Terminology, BC
Planning Life Cycle, Failure Analysis, Business Impact Analysis, BC Technology Solutions, Concept in Practice: EMC PowerPath.

Backup And Archive

Backup Purpose, Backup Considerations, Backup Granularity, Recovery Considerations, Backup Methods, Backup Architecture, Backup and Restore Operations Backup Topologies, Backup in NAS Environments, Backup Targets, Data Deduplication for Backup, Backup in Virtualized Environments, Data Archive, Archiving Solution Architecture,

Concepts in Practice- EMC NetWorker, EMC Avamar, and EMC Data Domain.

Unit - II

Objective: To impart knowledge about replication and concepts in practice like EMC SnapView.

Local Replication

Replication Terminology, Uses of Local Replicas, Replica Consistency, Local Replication Technologies , Tracking Changes to Source and Replica, Restore and Restart Considerations, Creating Multiple Replicas, Local Replication in a Virtualized Environment, Concepts in Practice: EMC TimeFinder, EMC SnapView, and EMC RecoverPoint.

Unit - III

Objective: To introduce basics of remote replication and concepts of Cloud Computing.

Remote Replication

Modes of Remote Replication, Remote Replication Technologies, Three -Site Replication, Data Migration Solutions, Remote Replication and Migration in a Virtualized Environment, Concepts in Practice: EMC SRDF, EMC MirrorView, and EMC Recover Point.

Cloud Computing

Cloud Enabling Technologies, Characteristics of Cloud Computing, Benefits of Cloud Computing, Cloud Service Models, Cloud Deployment Models, Cloud Computing Infrastructure, Cloud Challenges, Cloud Adoption Considerations.

Unit - IV

Objective: To bring awareness about security of storage infrastructure and concepts in practice like RSA and VMware.

Securing The Storage Infrastructure

Information Security Framework, Risk Triad, Storage Security Domains, and Security Implementations in Storage Networking, Securing Storage Infrastructure in Virtualized and Cloud Environments, Concepts in Practice: RSA and VMware Security Products.

Unit - V

Objective: To edify the basics of storage infrastructure management and the concepts in practice.

Managing The Storage Infrastructure

Monitoring the Storage Infrastructure, Storage Infrastructure Management Activities,Storage Infrastructure Management Challenges, Developing an Ideal Solution, Information Lifecycle Management, Storage Tiering, Concepts in Practice: EMC Infrastructure Management Tools.

Text Books:

1. G.Somasundaram, A.Shrivastava: EMC Corporation, Information Storage and Management: Storing, Managing and Protecting Digital Information in Classic, Virtualized and Cloud Environment, 2nd Edition, Wiley publication, 2012.
2. Robert Spalding, Storage Networks:The Complete Reference,1st Edition, Tata McGraw Hill/Osborne, 2003.

References:

1. Marc Farley, Building Storage Networks, 2nd Edition, Tata McGraw Hill/Osborne, 2001.
2. Meeta Gupta, Storage Area Network Fundamentals, 1st Edition, Pearson Education, 2002.

Web Resources:

1. http://www.snia.org/education/storage_networking_primer/stormngmnt

II Year MCA – IV Semester

MCA408 Artificial Intelligence And Neural Networks

Course Objective: ***To introduce the basic principles, techniques, and applications of Artificial Intelligence and to make learner familiar with basic principles of AI toward problem solving, inference, perception, knowledge representation, and learning.***

Unit – I

Objective: To introduce the basics of AI ,the techniques and production systems.

Introduction: AI problems, underlying assumption, AI Technique ,problems, problem spaces and search definition, a problem as a state space and search, production system characteristic, problem characteristic.

Unit - II

Objective: To impart knowledge about the search mechanisms like A*,Hill Climbing ,Predicate Logic and Prepositional Logic.

Searching

Searching for solutions, uniformed search strategies – Breadth first search, depth first Search. Search with partial information (Heuristic search),Hill Climbing, Greedy best first search, A* search, problem reduction-AND,OR,Graph,AO* algorithm, Predicate logic- Representation of simple fax, is a relationship, computable functions and predicates, resolutions-conversion to classform,basis of resolution, resolution in proportional logic, unification algorithm, resolution in predicate logic.

Unit - III

Objective: To introduce the concept of neural networks ,functional units and ANN.

Characteristics of Neural Networks

Historical Development of Neural Networks Principles, Artificial Neural Networks: Terminology, Models of Neuron, Topology, Basic Learning Laws, Pattern Recognition Problem, Basic Functional Units, Pattern Recognition Tasks by the Functional Units.

Unit - IV

Objective: To edify the basics of feedforward and feedback neural networks and their usage.

Feedforward Neural Networks

Introduction, Analysis of pattern Association Networks, Analysis of Pattern Classification Networks, Analysis of pattern storage Networks.

Feedback Neural Networks

Introduction, Analysis of Linear Auto associative FF Networks, Analysis of Pattern Storage Networks.

Unit - V

Objective: To introduce the concept of Pattern Recognition.

Competitive Learning Neural Networks & Complex pattern Recognition
Introduction, Analysis of Pattern Clustering Networks, Analysis of Feature Mapping Networks, Associative Memory.

Text Books:

1. Artificial Intelligence, 2nd Edition, E.Rich and K.Knight (TMH).
2. Artificial Intelligence – A Modern Approach. Second Edition, Stuart Russel, Peter Norvig, PHI/ Pearson Education.
3. Artificial Neural Networks B. Yagna Narayana, PHI

References:

1. Artificial Intelligence and Expert Systems – Patterson PHI.
2. Expert Systems: Principles and Programming- Fourth Edn, Giarrantana/ Riley, Thomson.
3. PROLOG Programming for Artificial Intelligence. Ivan Bratka- Third Edition – Pearson Education.
4. Neural Networks Simon Haykin PHI
5. Artificial Intelligence, 3rd Edition, Patrick Henry Winston., Pearson Edition.

Web Resources:

1. http://www.cin.ufpe.br/~tfl2/artificial-intelligence-modern-approach.9780131038059.25368.pdf
2. www.pearsonhighered.com/assets/hip/us/hip_us.../0136042597.pdf

II Year MCA – IV Semester

MCA409 Software Testing Methodologies

Course Objective: ***To understand different software testing techniques and strategies so as to use them in real time software testing.***

Unit - I
Objective: To introduce the basic idea of software testing techniques and strategies.

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 achievablepaths, path sensitizing, path instrumentation, application of path testing.

Unit - II
Objective: To understand what type of testing techniques should be used in software development and basic concepts of testing.

Transaction Flow Testing
Transaction flows, transaction flow testing techniques.

Dataflow testing
Basics of dataflow testing, strategies in dataflow 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
Objective: To enlighten the different products and expressions and introduce a logic based testing.

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
Objective: To introduce how to use state charts , state graphs and introduce the testing tips.

State, State Graphs and Transition testing
State graphs, good & bad state graphs, state testing, Testability tips.

Unit - V
Objective: To provide the different matrix of graph and node reduction.

Graph Matrices and Application
Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools.

Usage of JMeter and Win runner tools for functional / Regression testing, creation of test script for unattended testing, synchronization of test case, Rapid testing.

Text Books:

1. Software Testing Techniques – Boris Beizer, Dreamtech, second edition.

References:

1. The craft of software testing - Brian Marick, Pearson Education.
2. Software Testing Techniques – SPD(Oreille)
3. Software Testing in the Real World – Edward Kit, Pearson.
4. Effective methods of Software Testing, Perry, John Wiley.
5. Art of Software Testing – Meyers, John Wiley.
6. Software Testing Tools – Dr.K.V.K.K.Prasad, Dreamtech.

Web Resources:

1. www.cs.cmu.edu/~luluo/Courses/17939Report.pdf
2. www.ir.nmu.org.ua/bitstream/.../24a555d1011fa1e62f0374963b072e04.pdf
3. www.i2.informatik.rwth-aachen.de/dl/mbt08/lec_notes_04.pdf

II Year MCA – IV Semester

MCA410 Data Warehousing And Data Mining Lab

Course Objective: ***To understand how to use data warehousing tools, how to use data mining toolsthe basic principles, concepts and applications of data warehousing and data mining and the task of data mining as an important phase of knowledge recovery process.***

Using Weka and Informatica tools

1. Design a DFD with Simple Transformation, Make your own Assumptions by taking Any Schema of your choice?
2. Design a DFD taking into consideration the following Ports for Customer Schema: Cno, Cname, C location. By employing a FLAT FILE, and Insert any 5 Records in the Customer Schema?
3. Design a DFD taking into consideration the following Ports as Emp_Id, Ename,Salary, Dept_No and perform the following on Dept_No Port by grouping up the Salary Port by using Filter Transformation?
4. Calculate the TAX for the schema EMPLOYEE by considering the following:- Emp_Id, Ename, Salary, Dept_No by using Expression Transformation. Tax should be calculated on the Salary port with 20%
5. Consider the DEPT Schema with the ports Dept_No, Dname, Dlocation and perform XML Transformation?
6. Design a DFD taking into consideration the following EMP_Schema:Emp_Id, Ename, Deptno, Salary.Apply "Router transformation"
7. Design a DFD taking into Consideration the Employee Schema and take the Following ports: Emp_Id, Ename, Salary, Deptno and Sort on the port called Salary by using "Sorter Transformation"?
8. Design a DFD taking into consideration the following information EMP Table and DEPT Table and Join the Two Tables (EMP, DEPT) with a "Joiner Transformation"?
9. Design a DFD taking into consideration EMP_Schema and calculate the SUM & AVERAGE of Salary using Aggregator Transformation.
10. Demonstrations of association rule process on data set using apriori Algorithm.
11. Demonstrations on classification rule process on dataset student.arff, J48 algorithm.
12. Demonstrations of classification rule process on dataset employee.arrf using ID3 algorithm.
13. Demonstrations of classification rule process on dataset employee.arrf using Naïve Bayes algorithm.

14. Demonstrate clustering rule process on dataset student.arrf using single K-means.

15. Creation of simple transformation for employee data.

II Year MCA – IV Semester

MCA411 Web TechnologiesLab

Course Objective: ***The primary objective of the course is to learn web programming by designing and developing web based applications,their validations and deployment using HTML, XML, JavaScript , JavaBeans, Servlets, JDBC and JSP.***

List of Sample Programs

1. Write an HTML page including any required JavaScript that takes a number from one text field in the range of 0 to 999 and shows it in another text field in words. If the number is out of range, it should show "out of range" and if it is not a number, it should show "not a number" message in the result box.

2. Write an HTML page that contains a list of 5 countries. When the user selects a country, its capital should be printed next to the list. Add CSS to customize the properties of the font of the capital (color, bold and font size).

3. Write a calculator program in HTML that performs basic arithmetic operations (+,, /, * and %). Use CSS to change the foreground and background color of the values, buttons and result display area separately. Validate the input strings using JavaScript regular expressions. Handle any special cases like division with zero reasonably

4. 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.
   Home page, Registration and user Login, Books catalog, Shopping Cart, Payment By credit card, Order Confirmation.

5. Validate the Registration, user login, and payment by credit card pages designed in above program using JavaScript.

6. Write a servlet that takes name and age from an HTML page. If the age is less than 18, it should send a page with "Hello <name>, you are not authorized to visit this site" message, where <name>should be replaced with the entered name. Otherwise it should send "Welcome <name> to this site" message.

7. Implement the "Hello World!" program using JSP Struts Framework.

8. Create and save an XML document at the server, which contains 10 users information.

9. WAP to validate XML schema by including DTD.

10. Install Tomcat web server and Apache.

11. Jdbc connectivity using servlets
12. Write a JSP which does following job- inserts the details of 3 or 4 users who registers with the website by using registration form. Authenticate user by his password.

13. Create tables in database which contains the details of book such as Book name, price, quantity, amount. Modify the catalogue page in such a way that you should connect to the database and extract data from tables and display them in catalogue page using JDBC.

II Year MCA – IV Semester

MCA412 Mini Project

Course Objective: ***Students will learn to work as a team and to focus on getting a working project done on time with each student being held accountable for their part of the project.They can learn about and go through the software development cycle with emphasis on different processes - requirements, design, and implementation phases.They can understand programming language concepts. Students can apply theoretical knowledge by operating and optimizing software development.***

III Year MCA – V Semester

MCA501 Cryptography & Network Security

Course Objective: ***To impart knowledge in theories, principles and techniques of computer and network security. Students will learn basic cryptography, fundamentals of computer/network security, risks faced by computers and networks, security mechanisms, operating system security, secure systems design principles, and network security principles.***

Unit –I
Objective: To introduce the concepts providing security to the information passed through networks

Introduction :
Attacks, Services & Mechanisms, Security, Attacks, Security Services.

Conventional Encryption: Classical Techniques, Conventional Encryption Model, and Steganography, Classical Encryption Techniques.

Modern Techniques:
Simplified DES, Block Cipher Principles, DES Standard, DES Strength, Block Cipher Design Principles, Block Cipher Modes of Operation.

Unit –II
Objective: To understand the use of cryptography algorithms and protocols to achieve computer security.

Conventional Encryption Algorithms:
Triples DES, International Data Encryption Algorithm, RC4, CAST-128, Key Distribution, Random Number Generation, Placement of Encryption Function.

Unit –III
Objective: To know the key generation and its usage by applying in various algorithms.

Public Key Encryption
Principles of Public-Key Cryptosystems, RSA Algorithm, Diffie-Hellman Algorithm,Key Management, Fermat's & Euler's Theorem, Primality, the Chinese Remainder Theorem.

Unit –IV
Objective: To know the usage of Digital Signatures for passing the confidential information through networks.

Hash Functions:
Message Authentication & Hash Functions: Authentication Requirements, Authentication Functions, Message Authentication Codes, Hash Functions, Birthday Attacks, Security of Hash Function & MACS, MD5 Message Digest Algorithm, Secure Hash Algorithm (SHA).

Digital Signatures:
Digital Signatures, Authentication Protocol, Digital Signature Standard (DSS).

Unit –V
Objective: To know the security mechanisms to protect computer systems and networks.

Network & System Security:
Authentication Applications: Kerberos, X.509, Directory Authentication Service, Electronic Mail Security, Pretty Good Privacy (PGP), Security:Architecture ,Authentication Header,encapsulating security payloads,combining Secuirty associations, Key Management.

Web Security:
Secure Socket Layer & Transport Layer Security.

System Security:
Intruders, Viruses.

Text Books:

1. William Stallings, "Cryptography and Network Security: Principles and Practice", Fifth Edition, PEA, 2011.
2. Data Computer communications, Stallings, 7th Edition PEA,2004.

References:

1. Atul Kahate, "Cryptography and Network Security", TMH,2nd e,2008,
2. Data Communications, Gupta, Prentice Hall, 1st edition, 2011.
3. Network Security Essentials, William Stallings, 3rd edition, Pearson, 2007.

Web Resources:

1. Cryptography and Network Security Stallings
2. Cryptography and Network Security Forouzan
3. http://www.inf.ufsc.br/~bosco/ensino/ine5680/material-cripto-seg/2014-1/Stallings/Stallings_Cryptography_and_Network_Security.pdf
4. http://sharif.edu/~amini/files/stallings.pdf

III Year MCA –V Semester

MCA502 OOAD through UML

Course Objective:To provide a snapshot of the activities in the different phases of the object-oriented development life cycle.Model a real-world application by using object diagram,E-R and EER models,class diagram, Sequence Diagram ,Collaboration diagram and etc.

Unit –I

Objective: To bring out the language together and agree on a standard for laying out a way to visualize complex systems and their requirements

Introduction to UML:

The meaning of Object-Orientation, object identity, encapsulation, information hiding, polymorphism, genericity, importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture.

Unit –II

Objective: To understand the structure of a system by showing the system's *classes*, their attributes, operations (or methods), and the relationships among objects.To know the use of Object diagrams to render a set of objects and their relationships as an instance.

Basic structural Modeling:

Classes, relationships, common mechanisms, diagrams.

Advanced structural modeling:

Advanced relationships, interfaces, types & roles, packages, instances.

Class & object diagrams:

Terms, concepts, examples, modeling techniques, class & Object diagrams.

Unit –III

Objective: To know the relationships and interactions among software objects in the Unified Modeling Language (UML).To understand the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.

Collaboration diagrams

Terms, Concepts, depicting a message, polymorphism in collaboration diagrams, iterated messages, use of self messages.

Sequence diagrams

Terms, concepts, differences between collaboration and sequence diagrams, depicting synchronous messages with/without priority call back mechanism broadcast message.

Unit –IV

Objective: To understand the behavior – the set of all signals compatible with the system. An important feature of the behavioral approach is that it does not distinguish a priority between input and output variables.

Behavioral Modeling

Interactions, use cases, use case diagrams, activity diagrams.

Advanced Behavioral Modeling

Events and signals, state machines, processes & threads, time and space, state chart diagrams.

Unit –V

Objective: To know the interaction of volumes, or to get an idea of how they look from different angles and to explore ideas. They can be used to exhibit and sell a design to help visualize a design.

Architectural Modeling

Terms, concepts, examples, modeling techniques for component diagrams and deployment diagrams.

Text Books:

1. The Unified Modeling Language User Guide, Grady Booch,Rambaugh, Ivar Jacobson, PEA,First edition,1999.
2. Object Oriented Analysis & Design,Grady Booch, TMH,First Edition, 2005.

References:

1. Head First Object Oriented Analysis & Design, Mclaughlin, SPDOReilly, 2006.
2. The Unified Modeling Language Reference Manual, 2/e, Rambaugh, Grady Booch, etc., PEA,2005.
3. Object-Oriented Design with UML, Barclay, Savage, Elsevier, 2008.

Web Resources:

1. http://www.cs.umd.edu/~mvz/cmsc435-s09/pdf
2. http://www.dorsethouse.com/pdf/Dorset-House-fundood.pdf
3. http://www.matincor.com/documents/intro_ooad.pdf
4. http://pdfmanualdownload.hol.es/pdf/object-oriented-analysis-and-design-satzinger.pdf

III Year MCA –V Semester

MCA503 Visual Programming

Objective:
Understand the principles of graphical user interface design and develop desktop applications and web services using .NET

UNIT I
Objective: To impart knowledge about GUI and .Net basics

GUI concept – Data types – GUI Architecture – Message Processing – Keyboard and Mouse Handling Displaying Text and Graphics – File and Printer Handling – DDE – DDL – ODBC – COM/DCOM / CORBA - .NET Namespaces, Assemblies, .NET Memory Management, Process Management, Interoperation with COM.

UNIT II
Objective: To understand concepts about .Net framework

Transactions in .NET, Structures Exception Handling, Code Access Security, Web Controls using the .NET framework, The .NET Framework Class Library.

UNIT III
Objective: To understand concepts of VB.Net framework and use it in developing applications
VB.NET – basic features - Inheritance, Value Types, Operator Overloading, Exception Handling, Arrays and Collections, Properties, Delegates and Events, Windows Forms, Dialog Boxes and Controls, Graphical Output, Files, Data access.

UNIT IV
Objective: To understand concepts about C#.Net framework and use it in developing applications
C#.NET – basic features, Arrays and Collections, parameter arrays, inheritance, Garbage collection and Resource management.

UNIT V
Objective: To understand concepts about ASP.Net framework and use it in developing applications
ASP.NET – Validation controls – Accessing Data with web forms – Building ASP.NET applications – Building and XML web service handling XML.

Text Books

1. Jeff Prosise, Programming Microsoft .NET, Microsoft Press.
2. David S Plat, Introducing Microsoft .NET, 3rd Edition, Microsoft Press.

References:

1. Matt J. Crounch, "ASP.NET and VB. NET Web Programming", Pearson Education, 2006.
2. Kevin Hoffman, "Microsoft Visual C# 2005 Unleashed", Pearson Education, 2006.

Web Resources:

1. www.computer-pdf.com/**programming/visual**-basic
2. http://www.durhamtech.edu/academics/coursedescriptions/courseoutlines/CSC139.pdf

III Year MCA – V Semester

MCA504 E-Commerce

Course Objective: ***To introduce the concepts of electronic commerce and to understand how electronic commerce is used in business enterprises, governments, consumers and people in general.***

Unit –I
Objective: To impart knowledge about the basic concepts of E-Commerce and its applications.

Electronic Commerce: Framework, anatomy of E-Commerce applications, E-Commerce Consumer applications, E-Commerce organization applications. Consumer Oriented Electronic commerce, Mercantile Process models.

Unit –II
Objective: To introduce the concepts of electronic payment systems and EDI.

Electronic payment systems: Digital Token-Based, Smart Cards, Credit Cards, Risks in Electronic Payment systems.

Inter Organizational Commerce: Electronic Data Interchange(EDI), EDI Implementation, Value added networks.

Unit –III
Objective: To introduce the concepts of Intra Organizational Commerce and SCM.

Intra Organizational Commerce -Work Flow, Automation Customization and Internal Commerce, Supply chain Management.

Unit –IV
Objective: To impart knowledge about the Corporate Digital Library, Advertising and Marketing on Internet.

Corporate Digital Library:
Document Library, Digital Document types, corporate Data Warehouses.

Advertising and Marketing:
Information based marketing, advertising on Internet, on-line marketing process, market research.

Unit –V
Objective: To introduce the concepts of Consumer Search and Resource Discovery and Desktop video processing.

Consumer Search and Resource Discovery:
Information search and Retrieval, Commerce Catalogues, Information Filtering.

Multimedia: key multimedia concepts, Digital Video and electronic Commerce, Desktop video processing, Desktop video conferencing.

Text Books:

1. Frontiers of electronic commerce –Kalakota, Whinston, Pearson,First edition,2007.
2. Electronics Commerce,Chung H.Michael,PEA,First Edition 2000.

References:

1. Electronic Commerce – Gary P.Schneider –Thomson,7th Edition, 2007.
2. Electronic Commerce Tanban Efrain,PEA,First edition 2001.
3. E-Commerce fundamentals and applications Hendry Chan, Raymond Lee, Tharam Dillon, Ellizabeth Chang, John Wiley.

Web Resources:

1. www.tutorialspoint.com/e_commerce/e_commerce_tutorial.pdf
2. www.pearsonhighered.com/samplechapter/0131735160.pdf
3. www.wiley.com/college/turban/0471073806/sc/ch09.pdf

III Year MCA – V Semester

MCA505 Software Design Methodologies

Course Objective: It is intended to permit a systematic approach to designing large S/W systems. Their development follows a life-cycle composed of a set of phase identified as requirements specification, design, implementation, system testing and system evolution respectively.

Unit –I

Objective: To master attributes and assessment of quality, reliability and security of software design.

Basic concepts of Design:
Introduction, Characteristics of design activities

Essential elements of designs
Design Quality:
Software quality models: Hierarchical models, Relational models
The effect of design on software quality: efficiency, Correctness and reliability, Portability, Maintainability, Reusability, Interoperability

Quality attributes of software design:
Witt, Baker and Merritt's design objectives, Parnas and Weiss's requirements of good designs, Quality of development process

Unit –II

Objectives: To describe the different stages in this design process, principles of software development process, process selection regarding software Architecture.

Design Principles:
Basic rules of software design: Causes of difficulties, Vehicles to overcome difficulties, Basic rules of software design

Design processes:
The context of design in software development process, Generic design process: descriptive models, structure of software design methods

Software Architecture:
The general notion of architecture: The notion of software architecture: Prescriptive models, Descriptive models, Multiple view models, the roles of architecture in software design, software architectural style.

Unit –III

Objectives: To describe about software architectures, typical architectural styles, independent components and data abstraction

Description of Software Architectures:

The visual notation: Active and passive elements, Data and control Relationships, Decomposition/Composition of architectural elements

Typical Architectural Styles:

Data flow: The general data flow styles, the pipe- and filter sub-style, the batch sequential processing sub-style

Independent components:

The general independent components style, the event-based implicit invocation systems sub-style

Unit –IV

Objectives: To describe the stages and modeling of static attributes and the dynamic attributes of a system design.

Using Styles in Design:

Choices of styles, Combinations of styles, Hierarchical heterogeneous styles, simultaneously heterogeneous styles, locationally heterogeneous styles, Case Study: Keyword frequency vector: specification of the problem, designs in various styles, Analysis and comparison

Architectural Design space:

Theory of design spaces: Structure of design spaces, solving design synthesis and analysis problems, Design space of architectural elements: Behavior features, static features, Static features.

Design space of architectural styles:

Characteristic features of architectural styles, Classification of styles

Unit –V

Objectives: To analyze architectural design discover quality features by SAAM, quality attributes through ATAM and derivation of quality concerns.

Analysis and Evaluation: The concept of scenario, scenarios for evaluating modifiability: Scenarios for evaluating reusability, specification of operational profiles, evaluation and analysis of performance, Scenarios for evaluating reusability: Analysis and Evaluation of Modifiability: the SAAM Method: The input and output, the process (Activities in SAAM Analysis)

Quality Trade- Off Analysis:

The ATAM Method: ATAM analysis process, ATAM analysis activities

Model-Based Analysis:
The HASARD Method: Representation of quality models, construction of quality models, Hazard identification, Cause- consequence analysis, assembling graphic model, Identification of quality concerns

Derivation of quality features: contribution factors of a quality concern, sensitive quality attributes of a component, Quality risks, trade-off points.

Text Books:

1. Software Design Methodology: From Principles to Architectural Styles , Hong zhu, Elsevier,2009.
2. Software Architecture and Design, Bernard Witt, Baker, Merritt, Von Nostrand Reinhold,NY, 1994

References:

1. Software Architecture: Perspectives on an Emerging discipline, Shaw, M.,Garlan, PEA, 2008.
2. Software Architecture in Practice, Bass, L., Clements P,Kazman, PEA,2003
3. Evaluating Software Architectures: Methods and Case Studies, Clements, Kazman, Klien, PEA, 2002
4. Design and Use of Software Architectures- Adopting and Evolving a product – Line Approach, Bosch, J., ACM Press , Addison Wesley, 2000

Web Resources:

1. https://www.uio.no/studier/emner/matnat/ifi/INF9200/v10/readings/papers/Lowgren.pdf
2. http://www.dim.uchile.cl/~juaperez/beto/libro.guia.pdf
3. http://www.sei.cmu.edu/reports/89tr025.pdf
4. http://infolab.stanford.edu/~burback/watersluice/watersluice.pdf

III Year MCA – V Semester

MCA506 Design Patterns

Course Objective: To understand the design patterns and standard solutions to common s/w design problems able to use systematic approach that focus and describe abstract system of interaction between classes objects and communication flow.

Unit –I
Objectives: To add functionality to designs while minimizing complexity.

Introduction:
What Is a Design Pattern?, Design Patterns in Smalltalk MVC, Describing Design Patterns, The Catalog of Design Patterns, Organizing the Catalog, How Design Patterns Solve Design Problems, How to Select a Design Pattern, How to Use a Design Pattern.

Unit –II
Objectives: To refactoring the badly designed program properly using patterns

A Case Study:
Designing a Document Editor: Design Problems, Document Structure, Formatting, Embellishing the User Interface, Supporting Multiple Look-and-Feel Standards, Supporting Multiple Window Systems, User Operations Spelling Checking and Hyphenation, Summary.

Creational Patterns:
Abstract Factory, Builder, Factory Method, Prototype, Singleton, Discussion of Creational Patterns.

Unit –III
Objectives: To understand the common design patterns.

Structural Pattern Part-I:
Adapter, Bridge, Composite.

Structural Pattern Part-II:
Decorator, açade, Flyweight, Proxy.

Unit –IV
Objectives: To identifying the appropriate patterns for design problems.

Behavioral Patterns Part-I :
Chain of Responsibility, Command, Interpreter, Iterator.

Behavioral Patterns Part-II:
Mediator, Memento, Observer.

Unit –V

Objectives: To refactoring the badly designed program properly using patterns.

Behavioral Patterns Part-II (cont'd):

State, Strategy, Template Method, Visitor, Discussion of Behavioral Patterns. What to Expect from Design Patterns, A Brief History, The Pattern Community An Invitation, A Parting Thought.

Text Books:

1. Design Patterns by Erich Gamma, Pearson Education, 1995.
2. Design Patterns Explained By Alan Shalloway, Pearson Education.

References:

1. Pattern's in JAVA Vol - I By Mark Grand, Wiley DreamTech.
2. Pattern's in JAVA Vol-II By Mark Grand, Wiley DreamTech.
3. JAVA Enterprise Design Patterns Vol-III By Mark Grand, Wiley DreamTech.
4. Head First Design Patterns By Eric Freeman-Oreilly-spd

Web Resources:

1. https://sourcemaking.com/design_patterns/structural_patterns
2. https://sourcemaking.com/design_patterns/bridge
3. www.tutorialspoint.com/design_pattern/design_pattern_overview.htm
4. https://ece.uwaterloo.ca/~se464/08ST/lecture/06_design-patterns2.pdf
5. https://www.cs.umd.edu/class/spring2005/cmsc838p/.../designPatterns.pp.
6. https://www.pluralsight.com/courses/patterns-library

III Year MCA – V Semester

MCA507 Mobile Application Development

Course Objective: ***Students will gain a broad understanding of the discipline of Mobile Application Development using J2ME Technology.***

Unit- I
Objective: To gain the new web-centric in highly distributive systems to meet instaneous demand expected by concurrent users.

J2ME Overview:
Java 2 Micro Edition and the World of Java, Inside J2ME

J2ME and Wireless Devices Small Computing Technology:
Wireless Technology, Radio Data Networks, Microwave Technology, Mobile Radio Networks, Messaging, Personal Digital Assistants.

Unit- II
Objective: To understand J2ME architecture, Development environment and the reality of working.

J2ME Architecture and Development Environment:
J2ME Architecture, Small Computing Device Requirements, Run-Time Environment, MIDlet Programming, Java Language for J2ME, J2ME Software Development Kits, Hello World J2ME Style, Multiple MIDlets in a MIDlet Suite, J2ME Wireless Toolkit J2ME Best Practices and Patterns: The Reality of Working in a J2ME World, Best Practices

Unit- III
Objective: To gain the knowledge in J2ME application that you develop requires a way for a user to interact with it.

J2ME User Interface:
Commands, Items, and Event Processing J2ME User Interfaces, Display Class, The Palm OS Emulator, Command Class, Item Class, Exception Handling

High-Level Display:
Screens:
Screen Class, Alert Class, Form Class, Item Class, List Class, Text Box Class, Ticker Class

Low-Level Display:
Canvas:
The Canvas, User Interactions, Graphics, Clipping Regions, Animation

Unit- IV

Objective: To access and manipulate service-side components by web services and client-side applications.

J2ME Data Management:

Record Management System,Record Storage, Writing and Reading Records, Record Enumeration, Sorting Records, Searching Records

Record Listener JDBC Objects:

The Concept of JDBC, JDBC Driver Types, JDBC Packages, Overview of the JDBC Process, Database Connection, statement Objects, Result set, Transaction Processing, Metadata, Data Types

Exceptions JDBC and Embedded SQL:

Model Programs, Tables, Indexing, Inserting Data into Tables, Selecting Data from a Table, Metadata, Updating Tables, Deleting Data form a Table, Joining Tables, Calculating Data, Grouping and Ordering Data, Subqueries, VIEWs

Unit- V

Objective:To implement routines to open communications with other applications network connection and also learn to utilize web services and create large-scale distributive systems.

Generic Connection Framework:

The Connection, Hypertext Transfer Protocol, Communication Management Using HTTP Commands, Session Management, Transmit as a Background Process

Text Books:

1. J2ME: The Complete Reference, James Keogh, Tata Mc Graw- Hill Edition, 2003.
2. Wireless Java with J2ME,Morrison Michael,Tech Media,2001.

References:

1. Enterprise J2ME: Developing Mobile Java Applications –Michael Juntao Yuan, Pearson Education, 2004.
2. Beginning Java ME Platform, Ray Rischpater, Apress, 2009.
3. Beginning J2ME: From Novice to Professional, Third Edition, Sing Li, Jonathan B. Knudsen, A press, 2005.
4. Kicking Butt with MIDP and MSA: Creating Great Mobile Applications,1st edition, J.Knudsen,Pearson.

Web Resources:

1. http://www.vogella.com/articles/Android/article.html
2. http://androinica.com/category/androidguide/
3. http://www.learn-android-easily.com/

4. http://www.javatpoint.com/android-tutorial
5. http://www.raywenderlich.com/tutorials

III Year MCA –V Semester

MCA508 Software Project Management

Course Objective: ***To impart knowledge about specific roles within a software organization as related to project and process management ,Understanding the basic infrastructure competences (e.g., process modeling and measurement) and the basic steps of project planning, project management, quality assurance, and process management and their relationships .***

Unit –I
Objective: To introduce the concepts of different process models and evaluation and improving software economics through principles of conventional and modern software engineering methods.

Conventional Software Management:
The waterfall model, conventional software management performance.

Evolution of Software Economics:
Software Economics, pragmatic software cost estimation.

Improving Software Economics:
Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections.

The old way and the new way:
The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process.

Unit –II
Objective: To understand the phases in developing a product and the basic infrastructure competences.

Life cycle phases:
Engineering and production stages, inception, Elaboration, construction, transition phases.

Artifacts of the process:
The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts.

Unit –III
Objective: To understand the Management perspective and technical perspective in product development and the basic infrastructure competences.

Model based software architectures:
A Management perspective and technical perspective.
Work Flows of the process:

Software process workflows, Iteration workflows.

Checkpoints of the process
Major mile stones, Minor Milestones, Periodic status assessments

Iterative Process Planning
Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning.

Unit –IV
Objective: To understand the basic steps of project planning, project management and quality assurance.

Project Organizations and Responsibilities:
Line-of-Business Organizations, Project Organizations, evolution of Organizations.

Process Automation:
Automation Building blocks, The Project Environment.

Project Control and Process instrumentation
The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation.

Unit –V
Objective: To understand the relationship between future software management and modern process transitions.

Tailoring the Process:
Process discriminants.

Future Software Project Management:
Modern Project Profiles, Next generation Software economics, modern process transitions.

Text Books:

1. Software Project Management, Walker Royce, PEA, 2005.
2. Software Project Management in practice, PankajJalote, PEA,2005,

References:

1. Software Project Management, Bob Hughes,3/e, Mike Cotterell,TMH
2. Software Project Management, Joel Henry, PEA
3. Effective Software Project Management, Robert K.Wysocki,Wiley,2006
4. Project Management in IT, Kathy Schwalbe, Cengage

Web Resources:

1.http://walkerroyce.com/PDF/Successful_Software_Management.pdf
2.http://muele.mak.ac.ug/pluginfile.php/200806/mod_resource/content/1/Book%20Bhwalbe.pdf
3.http://gvpce.ac.in/syllabi/Software%20Project%20Managment.pdf
4.http://elibrary.com.ng/UploadFiles/file0_1618.pdf

III Year MCA – V Semester

MCA-509 Data Science

Course Objective: ***To understand the basic concepts of big data, methodologies for analyzing structured and unstructured data with emphasis on the relationship between the Data Scientist and the business needs.***

Unit- I:
Objective: To address the growing need for big data analytic skills.

Introduction foundations of Data Science.

Data Manipulation at Scale:
MapReduce, Hadoop, relationship to databases, algorithms, extensions, languages Databases , SQLand the relational algebra.

Unit-II:
Objective: To recognize and analyze ethical issues in business related to intellectual property, data security, integrity and privacy.

Parallel databases, parallel query processing, in-database analytics, Key-value stores and NoSQL; tradeoffs of SQL and NoSQL

Unit-III:
Objective: To use data mining software to solve real world problems

Statistical Analytics:
Programming in Python and R,Basic Data Mining-Basic statistical modeling, introduction to machine learning, over fitting Supervised learning-Linear and Logistic Regression, Classification, Unsupervised learning-Clustering, Association Rule mining

Unit-IV:
Objective: To apply quantitative modeling and data analysis techniques to the solution of business problems communicate finding and present results using data visualization techniques.

Graph/Text Data Analysis & Communicating Results:
Graph Analytics: Page Rank, community detection, recursive queries, iterative processing

Text Analytics:
TF/IDF, conditional random fields, Visualization, data products, visual data analytics.

Unit-V:

Objective: To apply ethical practices in every day business activities and make well reasoned ethical business and data management decisions.

Parallel Computing:

Concurrency and Data Decomposition, Message Based Parallelism –MPI, Thread Based Parallelism –OpenMP

Text Books:

1. An Introduction to Data Science by Jeffrey M. Stanton.
2. DJ Patil (16 September 2011). "Building Data Science Teams". O'Reilly Media, Inc.. Retrieved 7 July 2012.

References

1. Davenport, Thomas H.; Patil, D.J. (2012). "~~Data Scientist: The Sexiest Job of the 21~~st ~~Century". Harvard Business Review~~ October 2012: 70-76. Retrieved 13 October 2012.
2. Mike Loukides (June 2010). "What is Data Science?". O'Reilly Media, inc.. Retrieved 7 July 2012.
3. Longhow Lam (28 October 2010). "Introduction to R". PDF. The Comprehensive R Archive Network (CRAN). Retrieved 14 July 2012.
4. Emmanuel Paradis (12 September 2005). "R for Beginners". PDF. The Comprehensive R Archive Network (CRAN). Retrieved 14 July 2012.
5. Code School. "Try R". On-line Course. O'Reilly. Retrieved 16 May 2013.

Web Resources:

1. http://radar.oreilly.com/2010/06/what-is-data-science.html
2. http://www.nytimes.com/2009/08/06/technology/06stats.html
3. http://gerdleonhard.typepad.com/files/wef_ittc_personaldatanewasset_report_2011.pdf
4. http://radar.oreilly.com/2011/09/building-data-science-teams.html

III Year MCA –V Semester

MCA-510 UML Lab

Course Objective: *Main objective of this lab is to enable the student to practice the object - oriented analysis and design through UML on a particular application so that the student will apply same methodology in project which has to be done in VI semester. And also it will give exposure to tools that support UML and Object oriented software development.*

Suggested Applications

Exercise1. ATM APPLICATION.

Exercise 2. LIBRARY MANAGEMENT SYSTEM.

Exercise 3. ONLINE BOOK SHOP

Exercise 4. RAILWAY RESERVATION SYSTEM

Exercise 5. BANKING SYSTEM

Exercise6. AUCTION SYSTEM

Exercise7. POINT OF SALE SYSTEM

Exercise8. AIRLINE RESERVATION SYSTEM

II Year MCA – V Semester

MCA-511 Visual Programming Lab

Course Objective: ***To understand how to use Visual Programming concepts for the development of applications.***

List of Programs to be implemented using C#.Net,VB.Net and ASP.Net

1.Write a Program to Check whether a number is Palindrome or not.

2. Write a Program to demonstrate Command line arguments Processing.

3. Write a Program to find the roots of Quadratic Equation.

4. Write a Program to demonstrate boxing and unBoxing.

5. Write a Program to implement Stack operations.

6. Write a program to demonstrate Operator overloading.

7. Write a Program to find the second largest element in a single dimensional array.

8. Write a Program to multiply to matrices using Rectangular arrays.

9. Find the sum of all the elements present in a jagged array of 3 inner arrays.

10. Write a program to reverse a given string.

11. Using Try, Catch and Finally blocks write a program to demonstrate error handling.

12. Design a simple calculator using Switch Statement .

13. Demonstrate Use of Virtual and override key words with a simple program

14. Implement linked lists using the existing collections name space.

15. Write a program to demonstrate abstract class and abstract methods .

16. Write a program to build a class which implements an interface which is already existing.

17. Write a program to illustrate the use of different properties .

18. Demonstrate arrays of interface types .