ಕರ್ನಾಟಕ ರಾಜ್ಯ ಅಕ್ಕಮಹಾದೇವಿ ಮಹಿಳಾ ವಿಶ್ವವಿದ್ಯಾನಿಲಯ, ವಿಜಯಪುರ
(ಹಿಂದಿನ ಪದನಾಮ "ಕರ್ನಾಟಕ ರಾಜ್ಯ ಮಹಿಳಾ ವಿಶ್ವವಿದ್ಯಾನಿಲಯ, ವಿಜಯಪುರ")

Karnataka state Akkamahadevi Women's University, Vijayapura
(Formerly known as "Karnataka State Women's University, Vijayapura")

DEPARTMENT OF COMPUTER SCIENCE

Scheme of Teaching and Examinations and Syllabus of

Master of Computer Applications (MCA)

(Two Year Programme)

As per CBCS regulations

w.e.f. 2020-21 and onwards

Approved in BoS In Computer Science (PG) dated 16-09-2020

Scheme of Master of Computer Applications (MCA) programme, w.e.f. 2020-21 (CBCS regulations)

MCA I Semester (CBCS regulations) (w.e.f. 2020-21 and onwards)									
Course Code	Subject Title	Credits	Teaching Scheme Hrs/week			Examination			
						Continuous Assessment	Semester End examination		**Total Marks**
			Lecture	**Tutorial**	**Practical**		Exam. Duration (Hrs)	Theory/ Practical	
18MCAHC1.1	Data Structures	3:0:1	3	--	2	30	3	70	100
18MCAHC1.2	Operating System Principles with UNIX	3:0:1	3	--	2	30	3	70	100
18MCAHC1.3	Computer Networks	3:0:1	3	--	2	30	3	70	100
18MCAHC1.4	Discrete Mathematical Structures	3:1:0	3	2	--	30	3	70	100
18MCABC1.5	Web technologies-I	0:0:2	-	-	4	30	3	70	100
18MCASC1.6	Any One 1)Probability and Statistical Methods 2) Numerical Methods 3) Optimization Techniques 4) Linear Algebra	3:1:0	3	2	--	30	3	70	100
18MCABC1.7	Fundamentals of Programming**	0:0:0	2	--	2	30	3	70	100
18MCAOE1.8	**Open Elective** Offered by Dept. of Women's Studies	4:0:0	4	--	--	30	3	70	100
Total		**26**				210		490	**700**

***** HCT: Hard core theory HCP: Hard core Practical SCT: Soft core theory, OE: Open elective, BC: Bridge course

**Noncredit course, student has to obtain eligibility both in CA and SEE

The **BRIDGE COURSE 18MCABC1.7 entitled Fundamentals of Programming is a Non-Credit course for only B.Sc. / B.A. / B.Com with Mathematics at 10 + 2 Level or Graduation level.

MCA II Semester (new regulations under CBCS) (w.e.f. 2020-21 and onwards)									
Course Code	Subject Title	Credits	Teaching Scheme Hrs/week			Examination			
						Continuous Assessment	Semester End examination		Total Marks
			Lecture	Tutorial	Practical		Exam. Duration (Hrs)	Theory/ Practical	
18MCAHC2.1	Programming with Java	2:0:2	2	--	4	30	3	70	100
18MCAHC2.2	Algorithms analysis	3:1:0	3	2	0	30	3	70	100
18MCAHC2.3	Database Management System	3:0:1	3	--	2	30	3	70	100
18MCASC2.4	Artificial Intelligence	3:0:1	3	--	2	30	3	70	100
18MCASC2.5	Python programming	0:0:2	--	--	4	30	3	70	100
18MCASC2.6	Any one of the following a) Computer Graphics b) Cryptography and Network Security c) System Software d) Computer Architecture	3:1:0	3	2	--	30	3	70	100
18MCAOE27	**Open Elective** Offered by Dept. of Women's Studies	4:0:0	4	--	--	30	3	70	100
Total		**26**				210		490	**700**

***** HCT: Hard core theory HCP: Hard core Practical SCT: Soft core theory, OE: Open elective

MCA III Semester (new regulations under CBCS) (w.e.f. 2021-22 and onwards)									
Course Code	Subject Title	Credits	Teaching Scheme Hrs/week			Examination			
						Continuous Assessment	Semester End examination C3		Total Marks
			Lecture	Tutorial	Practical		Exam. Duration (Hrs)	Theory/ Practical	
18MCAHC3.1	Software Engineering concepts	3:1:0	3	2	--	30	3	70	100
18MCAHC3.2	Web Technologies-II	3:0:1	3	--	2	30	3	70	100
18MCAHC3.3	IOT	3:0:1	3	--	2	30	3	70	100
18MCAHC3.4	Data Analytics using Python	3:0:1	3	--	2	30	3	70	100
18MCASC3.5	Any One of the following a) Mobile Computing b) Machine Learning c) Cloud Computing d) Data Mining and Data Warehousing	3:0:1	3	--	2	30	3	70	100
18MCAHCP3.6	Practical-VI: Mini Project with IoT/Web Technology/Data Analytics.	0:0:2	--	--	4	30	3	70	100
18MCAOE3.7	**Open Elective (for other departments)** Web Design	4:0:0	4	--	--	30	3	70	100
Total		**26**				210		490	**700**

***** HCT: Hard core theory HCP: Hard core Practical SCT: Soft core theory, OE: Open elective

MCA IV Semester (new regulations under CBCS) (w.e.f. 2021-22 and onwards)									
Course Code	Subject Title	Credits	Teaching Scheme Hrs/week			Examination			
						Continuous Assessment/ Seminar	Semester End Examination C3		Total Marks
			Lecture (online classes)	Tutorial	Practical		Exam. Duration (Hrs)	Theory/ Practical/	
18MCAHCT6.1	Research Methodology	02	02	--	--	30	03	70	100
18MCAHCT6.2	Internship	02	--	--	--	--	--	100	100
18MCAHCT6.3	PROJECT WORK*	12	--	--	--	100	03	200	300
Total		**16**	--	--	--	130		370	**500**

* SEE comprises of Project report evaluation, presentation, and viva-voce in the ratio 50:25:25

All the students have to undergo compulsory Internship i.e. MOOCs certification (SWAYAM/NPTEL/COURSERA) of minimum 4/5 weeks duration during III/IV semester. Internship shall be considered as a head of passing and shall be considered for the award of degree. The MOOCs credits shall be included in IV semester score card.

PROJECT WORK - Instructions:

- It is a full time project to be taken up either in the Industry/Institute or in an R&D organization.
- The duration of the project is entire semester.
- Student has to carry out literature study pertaining to her project work which shall be assessed during CA and semester end Presentation.
- The student has to carry out the project work under the supervision of internal Guide(in the Department) and External Guide(Industry/Research Organization)
- The student shall submit the two copies of project work documentation to the department at the end of the semester.

Semester	Credits	Practicals in Hrs
MCA I	26	12
MCA II	26	12
MCA III	26	12
MCA IV	16	12

Total Credits: 94

Master of Computer Applications (MCA) programme has been designed to prepare graduates for attaining the following programme outcomes:

- Ability to apply knowledge of Mathematics, Computer Science and Management in practice
- Ability to identify, analyze, formulate and develop Computer Applications
- Ability to work with modern computing tools and techniques and use them with appropriate skills
- Ability to devise and conduct experiments, interpret data and provide well informed conclusions
- Ability to understand the impact of system solutions in a contemporary, global, economical, environmental, and societal context for sustainable development.
- Ability to design a computing system to meet desired needs within realistic constraints of an Industry/Organization/Institution
- Ability to function professionally with ethical responsibility as an individual as well as in multidisciplinary teams with positive attitude

Program Specific Outcomes for Master of Computer Applications (MCA):

1. Apply the theoretical concepts of Computer Science and Practical knowledge in analysis, design, and development of Computing systems and applications (software applications)
2. Work as a socially responsible professional by applying Computer Science Principles and Management practices.

Programme Pedagogy:

The various courses offered in the M.C.A. programme will have three major components such as Theory, Tutorial and Practical. Many courses have all the three components, some courses have only theory and tutorial/practical components and some courses have only theory components. So the nature of the course can be generally expressed as L:T:P model where L stands for lecture/theory, T stands for tutorial and P stands for practical.

The pedagogy to teach a particular course depends on the L:T:P model. In order to give a brief description about the pedagogy followed to teach a particular course, the courses with a particular L:T:P structure are grouped and the pedagogy followed to deliver the contents of the course is mentioned below:

For the courses that have theory and tutorial components (3:1:0), the theoretical concepts, principles and methods are explained with example analogy or use cases. Illustrative examples, theorem proving approaches, axioms, derivations, computing models and architectural descriptions are used to effectively demonstrate the ideas and to convey the philosophy of the course. Conventional black boards/white boards are used for writing and explanation. Smart boards, ICT tools such as power point, spreadsheet, word processing, database management and graphics are used for illustrations and descriptions of the concept. Animations, video clips and graphical illustrations are used whenever necessary to enhance the understanding of the concept. Group discussions, seminars and online demonstrations using specific tools are carried out to better understand the concepts.

For the courses that have theory and practical components (3:0:1), the theoretical concepts are taught as described in the previous paragraphs using conventional black/white board approach as well as smart ICT based approach. In addition, hands on experience will be provided through practical classes, where the students are allowed to use the computer and the related software tools to solve a particular problem, to provide a particular service as appropriate. With practical classes, students are exposed to current technology and gain an understanding how to solve a real time problem. A list of course specific assignments is used to practice and also to test the practical skills of the students.

For the courses that have all the three components i.e. Theory, Tutorial and Practical components, a blended mode of teaching, which includes conventional classroom teaching using black/white boards, smart classroom teaching using ICT tools, demonstrations through experiments and simulations followed by hands on experience with practical classes.

For the dissertation course, any real time/live projects will be selected and based on the nature of the project, field works for data collection, bridge courses for learning tools and technology needed to implement the solution to the problem undertaken are carried out. Further, internships at starts up companies/industries for more hands on experience with a particular platform will be encouraged.

MCA I Semester

20MCAHC1.1: Data Structures (3:0:1)

Course Outcome:

- Understand various types of data structures in solving a problem through programming.
- Able to identify the suitability of a particular data structure to solve a problem.
- Critically evaluate the efficient representation of data structures in the memory.
- Elucidate the various operations performed on a particular data structure.
- Understand and implement various sorting techniques

UNIT I

Data Structure Basic Concepts, Primitive and non-primitive data structures, Abstract Data Type, Introduction to Algorithms, Key features of an algorithm, Time Complexity, Space Complexity, Asymptotic Notations ,Big 'O', Notation , Best case Time Complexity, Average case Time Complexity, Worst case Time Complexity, Searching an element into an array-Linear Search and Binary Search. Linear and Non-Linear Data Structures.

UNIT II

Linked List- Singly Linked Lists. Single linked list operations-traversing, searching, inserting, deleting. Circular Linked Lists. Doubly Linked Lists.

Introduction to Stacks, Array Representation of Stacks, Operations on a Stack, Linked Representation of Stacks, Operations on a Linked Stack, Multiple Stacks, Reversing a Lis, Implementing Parentheses Checker, , Evaluation of Arithmetic Expressions, Recursion.

Introduction to Queues, Array Representation of Queues, Linked Representation of Queues, , Types of Queues, Applications of Queues.

UNIT III

Trees: Basic Terminology,, Types of Trees, Creating a Binary Tree from a General Tree, Traversing a Binary Tree, , Huffman's Tree, Applications of Trees.

Efficient Binary Trees-Binary Search Trees, Operations on Binary Search Trees. Threaded Binary Trees, AVL Trees, Red-Black Trees, Splay Trees.

Multi-way Search Trees-B Trees, B+ Trees, 2-3 Trees.

Heaps-Binary Heaps, Binomial Heaps.

UNIT IV

Graphs: Graph Terminology, Directed Graphs,Representation of Graphs, Graph Traversal Algorithms, Topological Sorting. Shortest Path Algorithms-Minimum Spanning Trees, Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Warshall's Algorithm.

UNIT V

Sorting and Hashing-Bubble Sort, Selection Sort, Quick Sort, Insertion Sort, Merge Sort, Radix Sort, shell sort, tree sort. Hashing Concepts, Hash functions: Division Method, Middle Square Method, Folding Method. Collision in Hashing.

References:

1. Data and File Structures using C, Thareja Reema, Oxford University Press New Delhi 2011.
2. Data Structures and Program Design in C++,Robert L. Kruse, Alexander J. Ryba, Prentice Hall
3. Data Structures Using C++, D.S. Malik, Course Technology, Cenage Learning.
4. Data Structures, Chitra, A Rajan, P T, Tata McGraw Hill, New delhi
5. Horowitz and Sahni, Fundamentals of Data Structures, W H Freeman & Co
6. Jean-Paul, Tremblay and Sorenson, An introduction to data structures with applications, McGraw-Hill

20MCAHC1.2: Operating System Principles with UNIX (3:0:1)

Course Outcome:

- Understand the fundamental concepts of the operating systems (OS).
- Understand and analyse theory and implementation of: processes, resource control (concurrency etc.), physical and virtual memory, scheduling, I/O and files.
- Demonstrate the working of basic commands of Unix environment and perform file processing.
- Demonstrate the usage of different shell commands, variable and AWK filtering to the given problem

UNIT I

Introduction to Operating Systems, Computer System Organization; Computer System Architecture; Operating System Operations; Computing Environments; Operating System Services; System Calls; System Programs; Operating System Structure; Virtual Machines; System boot.

UNIT II

Process Concept; Process Scheduling; Operations on Processes; Inter – Process Communication; Multi – Threaded Programming, Multithreading Models.

Process Scheduling: Basic Concepts, Scheduling Criteria, Scheduling Algorithms, Multiple Processor Scheduling. Process Synchronization, The Critical Section Problem: Peterson's Solution; Semaphores; Classical Problems of Synchronization

UNIT III

Deadlocks: System model; Deadlock Characterization, Methods for handling deadlocks; Deadlock Prevention; Deadlock Avoidance; Deadlock Detection and Recovery from deadlock.

Management Strategies, Swapping; Contiguous Memory Allocation; Paging; Segmentation; Virtual Memory Management; Demand Paging; Page Replacement; Allocation of Frames; Thrashing

UNIT IV

The File System: The File, The Parent-Child Relationship, The HOME Variable: The Home Directory, pwd, cd, mkdir, rmdir, Absolute Pathnames, Relative Pathnames, The Unix File System. The vi Editor: vi Basics, Input Mode, ex Mode and Command Mode.

Basic File Attributes: ls options, File Ownership, File Permissions, chmod, Directory Permissions, Changing the File Ownership More File Attributes: File Systems and Inodes, Hard Links, Symbolic Links, The Directory, umask, Modification and Access Times, find.

UNIT V

The Shell: The Shell's Interpretive Cycle, Shell Offerings, Pattern Matching-The Wild-cards, Escaping and Quoting, Redirection: The Three Standard Files, Two Special Files: /dev/null and /dev/tty, pipes, tee: Creating a Tee, Command Substitution

Creation, Internal and External Commands, Running Jobs in Background, Killing Processes with Signals, Job Control, at and batch, cron.

Essential Shell Programming: Shell Variables, Environment Variables, Shell Scripts, read, Using Command Line Arguments, exit and exit status of command, The Logical Operators, The if Conditional, using test and [] to Evaluate Expression, The case Conditional, expr, while: looping, for: looping with a list, set and shift, trap, Debugging Shell Scripts with set – x

References:

1. Sumitabha Das: UNIX Concepts and Applications, 4th Edition, Tata McGraw Hill, 2006.
2. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating Systems Principles, 8th Edition, Wiley – India.
3. UNIX: The Complete Reference: Kenneth Roson et al, Osborne/McGraw Hill
4. UNIX and Shell Programming: M G Venkateshmurthy, Pearson Education Asia, 2005
5. Behrouz A Forouzan and Richard F Gilberg
6. D M Dhamdhere: Operating Systems – A Concept Based Approach, 2nd Edition, Tata McGraw – Hill

20MCAHC1.3: Computer Networks (3:0:1)

Course Outcome:

- Apply the basic concepts of networking and to analyse different parameters such as bandwidth, delay, throughput of the networks for the given problem.
- Apply different techniques to ensure the reliable and secured communication in wired and wireless communication
- Analyse the networking concepts of TCP/IP for wired and wireless components
- Identify the issues of Transport layer to analyse the congestion control mechanism
- Design network topology with different protocols and analyse the performance using NS2

UNIT I
Applications, Requirements, Network Architecture, Implementing Network Software, Performance.

UNIT II
Perspectives on Connecting, Encoding (NRZ, NRZI, Manchester, 4B/5B), Framing, Error Detection, Reliable Transmission, Ethernet and Multiple Access Networks (802.3), Wireless.

UNIT III
Internetworking and Advanced Internetworking Switching and Bridging, Basic Internetworking (IP), Routing, The Global Internet, Routing among Mobile Devices.

UNIT IV
End-to-End Protocols and Congestion Control Simple Demultiplexer (UDP), Reliable Byte Stream (TCP), Queuing Disciplines, TCP Congestion Control, Congestion-Avoidance Mechanisms.

UNIT V
Network Security and Applications:
Cryptographic Building Blocks, Key Pre-distribution, Firewalls, Traditional Applications, Infrastructure Services.

References:

1. Computer Networks A Systems Approach, Larry L Peterson and Bruce S Davie, 5th Edition, MKP – 2012 – (1, 2 ,3.1,3.2,3.3, 3.4,4.1, 5.1,5.2 , 6.2,6.36.4, 8.1,8.2,8.5, 9.1,9.3)
2. James F. Kurose, Keith W. Ross, "Computer Networking – A Top-Down Approach Featuring the Internet", Fifth Edition, Pearson Education,
3. Nader. F. Mir, "Computer and Communication Networks", Pearson Prentice Hall Publishers
4. Ying-Dar Lin, Ren-Hung Hwang, Fred Baker, "Computer Networks: An Open Source Approach", Mc Graw Hill Publisher
5. Behrouz A. Forouzan, "Data Communication and Networking", Fourth Edition, Tata McGraw – Hill

20MCAHC1.4: Discrete Mathematical Structures (3:1:0)

Course Outcome:

- Understand the fundamentals of set theory and solve problems using set theory.
- Demonstrate critical thinking, analytical reasoning, and problem solving skills.
- Use propositional and predicate logic in knowledge representation and truth verification.
- Apply the concepts of functions and relations, counting techniques, algebraic structures and graph theory in problem solving.
- To understand and apply the fundamental concepts in graph theory

- To apply graph theory based tools in solving practical problems

UNIT I

Sets, relations and functions: Operations on sets, relations and functions, binary relations, partial ordering relations, equivalence relations, principles of mathematical induction

UNIT II

Mathematical Logic

Propositional Logic, Applications of Propositional Logic, Propositional Equivalences
Predicates and Quantifiers, Nested Quantifiers, Rules of Inference Introduction to Proofs

UNIT III

Combinatorics: Basic counting techniques: inclusion and exclusion, pigeon-hole principle, permutation, combination, summations. Introduction to recurrence relations and generating functions

UNIT IV

Algebraic structures and isomorphisms: Algebraic structures with one binary operation - semigroups, monoids and groups, congruence relation and quotient structures. Free and cyclic monoids and groups, permutation groups, substructures, normal subgroups. Algebraic structures with two binary operations - rings, integral domains and fields. Boolean algebra and Boolean ring.

UNIT V

Graph Theory

Graphs and Graphs models, Graph Terminology and Special Types of Graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths, Shortest-Path Problems, Planar Graphs, Graph Coloring

References:

1. Kenneth H Rosen, Discrete Mathematics and its Applications, McGraw Hill publications, 7th edition
2. J.K Sharma, Discrete Mathematics, Mac Millian Publishers India
3. Ralph P. Grimaldi: Discrete and Combinatorial Mathematics, 5th Edition, Pearson Education
4. C L Liu, Elements of Discrete Mathematics, Second Edition, Tata McGraw-Hill.
5. J P Tremblay and R Manohar, Discrete mathematical structures with applications to Computer Science, Tata McGraw-Hill

18MCASC1.6: Web technology (0:0:2)

Course Outcomes:

- Create Web page and to host own web site on internet.
- Learn about the protocols involved in internet technology

UNIT I

Introduction to WWW: Protocols and programs, secure connections, application and development tools, the web browser, What is server?, choices, setting up UNIX and Linux web servers, user login, dynamic IP , Web site design principles, planning the site and navigation.

UNIT II

Introduction to HTML : The development process, Html tags and HTML forms, web site structure Introduction to XHTML : XML, Move to XHTML, Meta tags, Character entities, frames and frame sets, inside browser.

UNIT III

Style sheets: Introduction to CSS, basic syntax and structure, using CSS, background images, colors and properties, manipulating texts, using fonts, borders and boxes, margins, padding lists, positioning using CSS, CSS2.

UNIT IV

JavaScript: Client side scripting, What is JavaScript, How to develop JavaScript, simple JavaScript, variables, functions, conditions, loops and repetition,

UNIT V

JavaScript and objects, JavaScript own objects, the DOM and web browser environments, forms and validations.

References:

1. Steven Holzner,"HTML Black Book" Dremtech press.
2. Web Technologies, Black Book, Dreamtech Press
3. Web Applications : Concepts and Real World Design, Knuckles, Wiley-India
4. Internet and World Wide Web How to program, P.J. Deitel & H.M. Deitel Pearson.

Practicals:

1. Design web pages of your department containing a description of the courses, faculties, library etc, use href, list tags.
2. Create class timetable using table tag.
3. Create user Student feedback form (use textbox, text area , checkbox, radio button, select box etc.)
4. Create a web page using frame. Divide the page into two parts with Navigation links on left hand side of page (width=20%) and content page on right hand side of page (width = 80%). On clicking the navigation Links corresponding content must be shown on the right hand side.
5. Use Inline CSS to format your resume that you created

6. Use External, Internal, and Inline CSS to format college web page that you created
7. Develop simple calculator for addition, subtraction, multiplication and division operation using JavaScript
8. Create HTML Page with JavaScript which takes three Integer number as input and tells which number is larger.
9. Create HTML Page that contains form with fields Name, Email, Mobile No , Gender , Favorite Color and a button now write a JavaScript code to combine and display the information in textbox when the button is clicked
10. Use regular expression for validation in Feedback Form

18MCASC1.6	**Any One** **a) Probability and Statistical Methods** **b) Numerical Methods** **c) Optimization Techniques** **d) Linear Algebra**

18MCASC1.6a: Probability and Statistical Methods (3:1:0)

Course Outcomes:

- Understand concepts of probability theory and statistical inference in order to solve applied problems.
- Familiarity with basic rules of probability and will be able to use them in modeling uncertainty in obtaining and recording data.
- Understand the logic of statistical inference and will be able to apply common inferential procedures

UNIT I

Probability: Sample Space, Events, Counting Sample Points, Probability of an Event, Additive Rules, Conditional Probability, Independence and Product Rules, Bayes' Rule

Random Variables and Probability Distributions: Concept of a Random Variable, Discrete Probability Distributions, Continuous Probability Distributions, Joint Probability Distributions

UNIT II

Mathematical Expectation: Mean of a Random Variable, Variance and Covariance of Random Variables, Means and Variances of Linear Combinations of Random Variables, Chebyshev's Theorem.

Some Discrete Probability Distributions: Introduction and Motivation, Binomial and Multinomial Distributions, Hypergeometric Distribution, Negative Binomial and Geometric Distributions 5.5 Poisson Distribution and the Poisson Process.

UNIT III

Some Continuous Probability Distributions: Continuous Uniform Distribution, Normal Distribution Areas under the Normal Curve, Applications of the Normal Distribution, Normal Approximation to the Binomial, Gamma and Exponential Distributions, Chi-Squared Distribution.

UNIT IV

Sampling Distributions and More Graphical Tools: Random Sampling and Sampling Distributions, Some Important Statistics, Sampling Distributions, Sampling Distribution of Means and the Central Limit Theorem, Sampling Distribution of S2, t-Distribution, F-Distribution.

UNIT V

One- and Two-Sample Estimation Problems: Introduction, Statistical Inference, Classical Methods of Estimation, Single Sample: Estimating the Mean, Standard Error of a Point Estimate, Prediction Intervals, Tolerance Limits, Single Sample: Estimating the Variance.

One- and Two-Sample Tests of Hypotheses: Statistical Hypotheses: General Concepts, Testing a Statistical Hypothesis, The Use of P-Values for Decision Making in Testing Hypotheses, Single Sample: Tests Concerning a Single Mean.

References:

1. Probability and Statistics for Engineers and Scientists, 9/E, by Walpole, Myers, Myers, Ye, Pearson 2012
2. H. Jeffreys, Theory of Probability, 3rd ed. Oxford: Oxford University Press, 1998
3. J.T. McClave, T. Sincich, A First Course in Statistics, 7th ed. Upper Saddle River, NJ: Prentice Hall; London: Prentice-Hall International
4. S.M. Ross, Introduction to Probability and Statistics for Engineers and Scientists,2nd ed. San Diego, CA; London: Harcourt/Academic,

18MCASC1.6b: Numerical Methods (3:1:0)

Course Outcome:

- Apply Numerical analysis which has enormous application in the field of Computer Science and Engineering
- Familiar with finite precision computation.
- Familiar with numerical solutions of nonlinear equations in a single variable. 4. Familiar with numerical integration and differentiation, numerical solution of ordinary differential equations. 5. Familiar with calculation and interpretation of errors in numerical method

UNIT I

Errors in Numerical Calculation – Introduction, Numbers and their Accuracy, Mathematical Preliminaries, errors and their computation ,absolute, relative and percentage errors, General error formula, Error in the series Approximation.
Solving Non linear Equations - computer & arithmetic errors, method of bisection, the secant method, Newton–Raphson method, Newton's method for polynomial, Horner's method, Muller's method, order of convergence of other method.

UNIT II
Interpolation- Introduction, errors in polynomial Interpolation, Finite Difference, Forward, Backward, Central Difference, Newton's Formulae for Interpolation, Lagrange's Interpolation Formula.
Linear System of Equation- Matrix notation, determinants and matrix inversion, norms, eigen values and eigen vectors of a matrix, the elimination method, Gauss elimination and Gauss-Jordan Method, Iterative method Jacobi Iterative Method and Gauss Seidal Iteration Method.

UNIT III
Curve Fillting, B- Splines and Approximation - Least –Square Curve Fitting procedures, fitting a straight line, nonlinear curve fitting, Method of Least Squares for continuous Functions, Orthogonal Polynomial , Gram-Schmidt Orthogonalization Process, B-Splines, Least Square solution, Representation of B- Spines, The Cox-de Boor Recurrence Formula, Computation of B-Splines, Approximation of functions, Chebyshev polynomials, Economization of Power Series.

UNIT IV
Numerical Differentiation – Errors in Numerical Differentiation, The cubic spline method, Maximum and Minimum values of a Tabulated function.
Integration - Numerical Integration, Trapezoidal Rule, Simpson's 1/3 Rule, Simpson's 3/8 Rule, Boole's and Weddle's Rulres, Romberg Integration, Newton-Cotes Integration Formulae, Euler – Maclaurin Formula.

UNIT V
Numerical Solution of Ordinary Differential Equations- Solution by Taylor's Series, Picard's Method of successive approximation, Euler'e Method, Error Estimate for the Euler Method, Modified Euler's Method, Rung – Ktta Method , Preditor- Corrector Methods, Adams-Moulton Method , Milne;s Method, Boundary Value Problems, Finite Difference Method, The Shooting Method, the cubic spline Method.

References

1. S.S.Satry – Introductory Methods of Numerical Analysis, 3rd edition Prentice-Hall India
2. M.K.Jain, S.R.K.Iyengar, R.K.Jain- Numerical Methods for Scientific and Engineering Computation, 3rd Edition New Age International (P) Limited.
1. F. Gerald, Patrick O. Wheatley, Applied Numerical Analysis, 6/e, Pearson Education.
2. Madhumangal Pal, Numerical Analysis for Scientists and Engineers, Narosa Publications.
3. Conte S.D. and Carl DeBoor, Elementary Numerical Analysis, McGraw Hill .
4. Shankar Rao K., Numerical Methods for Scientists and Engineers, PHI.

18MCASC1.6c: Optimization Techniques (3:1:0)

Course Outcome:

- Understand importance of optimization of industrial process management
- Apply basic concepts of mathematics to formulate an optimization problem
- Analyse and appreciate variety of performance measures for various optimization problems

UNIT I

Introduction: Operations research model, solving the OR model, art of modeling, phases of OR study.

Linear Programming: Formulation of a LP model, graphical solution to LPP, LP applications, standard and canonical form of LPP, the simplex method, big M-method, two-phase simplex method, special cases in simplex method, sensitivity analysis.

UNIT II

Duality and Post-Optimal analysis: Definition of dual problem, primal-dual relationships, economic interpretation of duality, dual simplex method, generalized simplex algorithm, post-optimal analysis.

UNIT III

Transportation models: Definition of transportation model, the transportation algorithm, the assignment model-the Hungarian method; the transshipment model.

UNIT IV

Network Models: Scope and definition of Network models, minimal spanning tree algorithm, shortest-route problem, maximal flow model, CPM and PERT.

UNIT V

Advanced Linear Programming: Simplex method fundamentals, revised simplex method, bounded-variable algorithm, duality, parametric linear programming.

Decision Analysis and Game theory: Decision making under certainty, decision making under risk, decision under uncertainty, optimal solution of two-person zero-sum games, solution of mixed strategy games.

References:

1. Hamdy A. Taha, Operations Research, 8/e, Pearson Education.
1. Panneerselvam R., Operations Research, PHI.
2. Gillet B.E, Introduction to Operations Research, TMH.
3. Sharma J.K, Operations Research, Theory and Applications, McMillan India Ltd.

18MCASC1.6d: Linear Algebra (3:1:0)

Course Outcome

- Solve systems of linear equations.
- Use matrix and vector algebra, and relate matrices to linear transformations. Understand and apply fundamental concepts of vector spaces such as span, linear independence, basis, and dimension.
- Apply the matrix calculus in solving a system of linear algebraic equations.
- Computing determinants
- Finding characteristic polynomial, eigenvalues and eigenvectors of matrices and linear transformation.

UNIT I

Systems of linear equations and matrices:

Systems of homogeneous and non-homogeneous linear equations- The solutions of systems of m homogeneous linear equations in n unknowns by elimination, The existence of non-trival solution of such a system for m< n . The sum of two solutions and a scalar multiple of a solution of such a system is again a solution of the system,.

Matrices over R, The matrix representation of systems of homogeneous and non-homogeneous linear equations. Addition, scalar multiplication and multiplication of matrices, Transpose of a matrix, types of matrices, Transpose of product of matrices, Invertible matrices, Product of invertible matrices.

Elementary row operations on matrices, row echelon from of a matrix and Gaussian elimination method. Applications of Gauss elimination method to solve system of linear equations

UNIT II

Vector spaces over R

Definition of a vector space over R and example, Subspaces – definition and examples, The sum and intersection of subspaces, direct sum of a subset of a vector space, Linear combination of vectors, convex sets, linear span of subset of a vector space, Linear dependence and independence of a set.

Basis of a vector space, basis as a maximal linearly independent set and a minimal set of generators. Dimension of a vector space, Row space, Column space of an m n × matrix over R and row rank, column rank of a matrix, Equivalence of row rank and column rank, Computing rank of a matrix by row reduction.

UNIT III

Inner Product Spaces

Dot product in n R , Definition of general inner product on a vector space over R . Norm of a vector in an inner product space. Cauchy-Schwarz inequality, triangle inequality. Orthogonality of vectors, Pythagorus theorem and geometric applications in R^2. Orthogonal complements of a subspace, Orthogonal Complements in R^2 and R^3. Orthogonal sets and orthonormal sets in an inner product space. Orthogonal and orthonormal bases.

UNIT IV

Linear Transformations

Linear transformations – definition and properties, The sum and scalar multiple of linear transformations from U to V where U, V are finite dimensional vector spaces over R is again a linear transformation. The space L U V (,) of linear transformations from U to V. Kernel and image of a linear transformation. Rank-Nullity Theorem. The linear isomorphisms, inverse of a linear isomorphism. Matrix of sum of linear transformations and scalar multiple of a linear transformation. Matrices of composite linear transformation and inverse of a linear transformation. Equivalence of rank of an m n × matrix A and rank of the linear transformation. The solutions of non-homogeneous systems of linear equations represented by AX= B

UNIT V

Determinants, Eigen Values and Eigen Vectors

Definition of determinant as an n-linear skew-symmetric function. Determinant of a matrix as determinant of its column vectors (or row vectors). Existence and uniqueness of determinant function via permutations. Computation of determinant of 2x2, 3x 3 matrices, diagonal matrices. Basic results on determinants. Laplace expansion of a determinant, Vandermonde determinant, determinant of upper triangular and lower triangular matrices. Linear dependence and independence of vectors in R^n using determinants. The existence and uniqueness of the system AX= B.

Eigenvalues and eigenvectors- Eigenvalues and eigenvectors of a linear transformation. Eigenvalues and eigenvectors of nX n real matrices and eigenspaces. The characteristic polynomial of an n n × real matrix, characteristic roots.

References:

1. Serge Lang : Introduction to Linear Algebra, Springer Verlag.
2. S. Kumaresan : Linear Algebra A Geometric approach, Prentice Hall of India Private Limited
3. M. Artin : Algebra, Prentice Hall of India Private Limited
4. K. Hoffman And R. Kunze : Linear Algebra, Tata McGraw Hill, New Delhi.
5. Gilbert Strang : Linear Algebra and its applications, International Student Edition

20MCABC1.7: Bridge Course: Fundamentals of Programming

Course Outcome

- To introduce the fundamental concepts of computers and computing environment.
- To acquire the basic knowledge of algorithm design and problem solving using c.
- To understand the concept of object oriented programming and acquiring skills for problem solving using OOPs syntax.

Assessment Method:

Written: Homework assignments, internal assessment test in class, and project work completion in home.
Laboratory: problem solving -writing and executing computer programs.
Tutorials: which allow for exercises in problem solving and allow time for students to resolve problems in understanding of lecture material.
Verbal: Classroom exercises, seminars and discussion and small periodic quizzes, to enable to assess students understanding of the concepts.

UNIT I
Digital Computers and Digital system: Number systems, Number base conversion, Complements, Binary codes, Binary arithmetic's. Boolean algebra: definitions, Basic theorems and properties of Boolean algebra, Venn diagram. Fundamentals of Computers. Introduction to Operating System.

UNIT II
Problem solving techniques: Problem solving procedure. Algorithm: Steps involved in algorithm development, Algorithms for simple problems, Flowcharts, Pseudocode. Introduction to C: Overview of C Program, Basic structure of a C - program. Constants, Variables & Data types: Character set, C token, Keywords & identifiers.

UNIT III
Control Structures - Conditional and Iterative Statements, Program Decomposition and Functions. 1-D and 2-D Arrays, Pointers, Dynamic Memory Allocation, and C Strings, structures and unions, File I/O.

UNIT IV
Object Oriented Programming Using C++: Classes objects, data members, member functions, this Pointer, Friends, Friend Functions, Friend Classes, Friend Scope, and Static Functions. Overloading Unary Operators, Overloading binary operators.

UNIT V
Types of Inheritance, Pointers, Objects and Pointers, Multiple Inheritance. Polymorphism, Abstract classes. Character and String input and output to files, STL, Command Line Arguments,

References:

1. M. Morris Mano, Digital Logic and Computer design, PHI, 2015
2. Thomas. C. Bartee, Digital Computer Fundamentals, 6th edition, TMH
3. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice Hall of India
4. E. Balagurusamy, Programming in ANSI C, 7th Edition, Tata McGraw Hill
5. Object Oriented Programming with C++ by E. Balagurusamy
6. C++ programming cookbook, Herb Schildt's ,_McGraw-Hill

Continuous Assessment Scheme

C1: Test : 10 marks
Seminar/assignment : 05 marks

C2: Test : 10 marks
Seminar/assignment : 05 marks

Semester End Examination (C3)

Theory Question Paper Pattern

Time: 3 Hrs. Max . Marks: 70

- There shall be one compulsory question consisting of 5 sub questions each of 2 marks each.
- There shall be seven questions of 12 marks each.
  - Each question may have sub questions (a),(b) / (a),(b),(c)
- The student has to answer any five full questions out of remaining seven questions

MCA II Semester

20MCAHC2.1: Programming with Java (2:0:2)

UNIT I

Introduction to Java: Java Architecture and Features, Compiling and Executing a Java Program, Variables, Constants, Keywords Data Types, Operators and Expressions, Decision Making Constructs and Nesting, Java Methods.

Arrays, Strings and I/O: Creating & Using Arrays, Referencing Arrays Dynamically, Java Strings: The Java String class, Creating & Using String Objects, Manipulating Strings, String Immutability & Equality, Passing Strings To & From Methods, String Buffer Classes. Simple I/O using System.out and the Scanner class, Byte and Character streams, Reading/Writing from console and files.

UNIT II

Principles of Object-Oriented Programming, Defining & Using Classes, Controlling Access to Class Members, Class Constructors, Method Overloading, Class Variables & Methods, Objects as parameters, final classes, Object class, Garbage Collection.

Inheritance: Single Level and Multilevel, Method Overriding, Dynamic Method Dispatch, Abstract Classes, Interfaces and Packages, Extending interfaces and packages, Package and Class Visibility, Using Standard Java Packages (util, lang, io, net), Wrapper Classes, Autoboxing/Unboxing, Enumerations and Metadata.

UNIT III

Exception Handling - Exception types, uncaught exceptions, throw, built-in exceptions, Creating your own exceptions;

Multi-threading: The Thread class and Runnable interface, creating single and multiple threads, Thread prioritization, synchronization and communication, suspending/resuming threads.

UNIT IV

Java Applets: Introduction to Applets, Writing Java Applets, Working with Graphics, Incorporating Images & Sounds. Event Handling Mechanisms, Listener Interfaces, Adapter and Inner Classes.

The design and Implementation of GUIs using the AWT controls, Swing components of Java Foundation Classes such as labels, buttons, text fields, layout managers, menus, events and listeners; Graphic objects for drawing figures such as lines, rectangles, ovals, using different fonts.

UNIT V

Connecting to Databases – Basic steps of JDBC - Databases and SQL - Retrieving Information - Storing Information - Accessing Multimedia databases – Working with database metadata - Database support in Web applications.

Java Socket and URLs - Socket and Interprocess Communication - Client/Server Methodology - Content and Protocols handlers – Developing distributed applications – CORBA and IIOP - Interfaces - RMI – Remote objects – Object serialization.

References:

1. Ken Arnold, James Gosling, David Homes, "The Java Programming Language", .
2. James Gosling, Bill Joy, Guy L Steele Jr, GiladBracha, Alex Buckley"The Java Language Specification, Java SE 8 Edition (Java Series)", Published by Addison Wesley, 2014.
3. Herbert Schildt, "Java The Complete Reference", Eighth Edition, Tata Mc Graw-Hill Edition India, 2011.
4. Cay S. Horstmann, GaryCornell, "Core Java 2 Volume 1 ,9th Edition, Prentice Hall.2012
5. Cay S. Horstmann, Gary Cornell, "Core Java 2 Volume 2 - Advanced Features)", 9th Edition, Printice Hall.2013
6. E. Balaguruswamy, "Programming with Java", 4th Edition, McGraw Hill.
7. Paul Deitel, Harvey Deitel, "Java: How to Program", 10th Edition, Prentice Hall, 2011.

20MCAHC2.2: Algorithms analysis (3:1:0)

20MCAHC2.3: Database Management System (3:0:1)

Course Outcome

- Understand the basic concepts of database systems, database transactions and related database facilities like concurrency control, data object locking and protocols
- Analyze the database requirements and determine the entities involved in the system and their relationships to one another
- Develop the logical design of the database using data modelling techniques
- Create a relational database and use data manipulation language to query, update and manage a database

UNIT I

Introduction to Database system concepts: Data models, schemas and instances, DBMS architecture and data independence, Database languages and interfaces, database system environment, Classification of DBMS.

Disk storage, basic file structures and hashing: Secondary storage devices, buffering of blocks, Placing File Records on Disk Operations on Files, Files of Unordered Records, Files of Ordered Records hashing techniques.

UNIT II

Data modelling using ER model : Entities, attributes and relationships, Different types of attributes, E- R Diagrams, Specialization and generalization, constraints and characteristics of specialization and generalization, Relationship types of degree higher than two.

Relational Data Model and Database design: Relational Model Concepts, Relational Model Constraints and Relational Database Schemas, Update Operations, Transactions, and Dealing

with Constraint Violations. Relational database design using ER to Relational Mapping, Mapping EER Model concepts to relations.
Database Design: Informal design guidelines for Relation schemes, Functional dependencies, Normal forms based on primary keys, General definitions of second and third normal forms.
Boyce – Code normal form, multi-valued dependencies and fourth normal form, Join dependencies and fifth normal form.

UNIT III
SQL : SQL Data Definition and Data Types, Specifying Constraints in SQL, Basic Retrieval Queries in SQL, INSERT, DELETE, and UPDATE Statements in SQL, Additional features of SQL. Complex Queries, Triggers, Views, and Schema Modification More Complex SQL Retrieval Queries, Specifying Constraints as Assertions and Actions as Triggers, Views (Virtual Tables) in SQL, Schema Change Statements in SQL.

UNIT IV
Transaction Management: Transaction - Introduction to transaction processing, transaction and system concept, Desirable properties of transaction, Transaction support in SQL, concurrency control techniques – Two phase Locking techniques for concurrency, timestamp based protocol.

UNIT V
Distributed Database: Distributed database concepts, Types of Distributed Database Systems, Data Fragmentation, Replication, and Allocation Techniques for Distributed Database Design.
Object, object relational and XML database: Object and Object-Relational Database– Overview of Object Database Concepts, Object- Relational Features: Object Database Extensions to SQL, The ODMG Object Model and the Object Definition Language ODL, The Object Query Language OQL.

References:

1. Elmasri & Navathe, *Fundamentals of Database Systems*, Addison-Wesley, 6th Edition, 2010.
2. Korth F. Henry and Silberschatz Abraham, *Database System Concepts*, McGraw Hill, 6th Edition, 2010.
3. O'neil Patric, O'neil Elizabeth ,*Database Principles, Programming and Performance*, Argon Kaufmann Publishers, 2nd Edition, 2002.
4. Ramakrishnan and Gehrke, *Database Management System*, McGraw-Hill, 3rd Edition, 2003

20MCAHC2.4: Artificial Intelligence (3:0:1)

Course Outcome:

- To understand basic principles of Artificial Intelligence
- Understand formal methods of knowledge representation, logic and reasoning

- Understand foundational principles, mathematical tools and program paradigms of artificial intelligence
- Design an application of artificial intelligence (AI)

Unit I Automated Reasoning - foundations of knowledge representation and reasoning, representing and reasoning about objects, relations, events, actions, time, and space; predicate logic, situation calculus, description logics, - Logic - Propositional and predicate logic - Syntax - Informal and formal semantics - Equivalence - De Morgans laws - Decidable problems - Many-sorted logic - first-order, higher-order logic- Reasoning methods - Formal program techniques - pre- and post-conditions, derivation and verification of programs -SPIN Tool.

Unit II Uncertain Knowledge - Bayesian networks; Basics of decision theory, sequential decision problems, elementary game theory; Problem-solving through Search - forward and backward, state-space, blind, heuristic, problem-reduction, A, A*, AO*, minimax, constraint propagation, neural and stochastic; Introduction to intelligent agents; Machine Learning - Foundations of supervised learning - Decision trees and inductive bias, Regression Vs Classification, Supervised - Linear Regression, Logistic Regression.

Unit III Generalisation, Training, Validation and Testing, Problem of Overfitting, Bias vs Variance ,Confusion Matrix, Precision, Recall, F Measure, Support Vector Machine, Decision Tree, Random Forest, Perceptron, Beyond binary classification, Boosting and bagging, bootstrapping - Advanced supervised learning - K-Nearest Neighbour, Markov model, Hidden Markov Model - Nearest Neighbor Classification - Gaussian processes - Unsupervised Learning - Dimensionality Reduction Techniques, Linear Discriminant Analysis - Clustering: K-means, Hierarchical, Spectral ,subspace clustering, association rule mining.

Russell, Norvig, Arti_cial Intelligence: A Modern Approach, Third edition, Prentice Hall, 2010
2. Hastie, Tibshirani, Friedman. The elements of statistical learning, Second edition, Springer, 2009
3. Tsang. Foundations of constraint satisfaction, Academic press, 1993
4. Daphne Koller and Friedman. Probabilistic Graphical Models - Principles and Techniques, The MIT Press, 2009

20MCAHC2.5: Python programming (0:0:2)

Course Outcome

- To acquire programming skills in core Python.
- To acquire Object Oriented Skills in Python.
- To develop the skill of designing Graphical user Interfaces in Python.
- Demonstrate significant experience with the Python program development environment.
- Understand and implement python modules like NumPy, Tkinter, Matplotlib

UNIT I

Introduction to Python: Structure of Python Program-Underlying mechanism of Module Execution-Branching and Looping-Problem Solving Using Branches and Loops-Functions Lists and Mutability- Problem Solving Using Lists and Functions

UNIT II

Sequence Data types and Object-Oriented Programming:

Sequences, Mapping and Sets- Dictionaries- -Classes: Classes and Instances-Inheritance-Exceptional Handling-Introduction to Regular Expressions using "re" module.

UNIT III

Using NumPy: Basics of NumPy-Computation on NumPy-Aggregations-Computation on Arrays-Comparisons, Masks and Boolean Arrays-Fancy Indexing-Sorting Arrays-Structured Data: NumPy's Structured Array.

UNIT IV

GUI Programming: Tkinter overview-Adding buttons and callbacks-user defined call handlers-top level windows-Dialogs-Check button, Radio button and scale-Menus-canvas-Grids

UNIT V

Visualization and MATPLOTLIB: Basic functions of matplotlib-Simple Line Plot, Scatter Plot-Density and Contour Plots-Histograms, Binnings and Density-Customizing Plot Legends, Colour Bars-Three-Dimensional Plotting in Matplotlib

References:

1. Zhang.Y, *An Introduction to Python and Computer Programming*, Springer Publications, 2016
2. Mark Lutz, *Programming Python*, O'Reily Media Inc., 2013
3. Jake VanderPlas, *Python Data Science Handbook - Essential Tools for Working with Data*, O'Reily Media Inc., 2016
4. T.R.Padmanabhan, *Programming with Python*, Springer Publications,2016

18MCASC2.6	**Any one of the following** **a) Computer Graphics** **b) Cryptography and Network Security** **c) System Software** **d) Computer Architecture**

18MCASC2.6a: Computer Graphics (3:1:0)

Course Outcome:

- Understand the basic concepts of computer graphics.
- Learn the implementation of algorithms to draw a line, circle, polygon, colour the objects, clipping the text and the object.
- Understand and implement the algorithms for 2D and 3D transformations.
- Learn the importance of viewing and projections.
- Understand the fundamentals of animation, virtual reality and its related technologies.
- Understand a typical graphics pipeline and implement the related algorithms.

UNIT I

Introduction: Survey of computer Graphics and its applications; Interactive and passive graphics; A graphics system: Video display devices, raster scan and random scan system, The synthetic camera model; The programmer's interface; Graphics architectures; Programmable pipelines; The OpenGL API, Primitives and attributes, Color Models – RGB, YIQ, CMY, HSV; Index color model; Viewing functions; Control functions; Graphics Programming: The Sierpinski gasket.

UNIT II

Interactive Graphics; Input devices: physical input devices and logical input devices, Clients and servers Model; Display lists; Graphics modeling using Display lists, Programming event driven input; Menus; Building animating interactive models. Geometry: Scalars, points, and vectors; Three-dimensional primitives; Coordinate systems and frames; Modeling a colored cube; 3-D Geometric transformations: Translation, rotation, scaling, reflection and shear transformations; Matrix representations and Homogeneous coordinates; Concatenation of transformations; OpenGL transformation matrices.

UNIT III

Viewing: Classical and computer viewing; Viewing with a computer; Positioning of the camera; Introduction to projections; Projections in OpenGL; Classifications of Projections; Parallel-projection ; Perspective-projection; Deriving Matrices for Parallel and Perspective Projections; Projections and shadows. Clipping and Rasterization; Clipping; Line-segment clipping Algorithms: Cohen–Sutherland algorithm, Liang–Barsky algorithm; Polygon clipping: Sutherland–Hodgman algorithm; Text Clipping; Rasterization;

UNIT IV

Line Drawing algorithms: Digital Differential Analyzer (DDA) algorithm, Bresenham's algorithm; Circle Drawing algorithm, Polygon rasterization: Scan line polygon fill algorithm, boundary-fill and flood-fill algorithms; Hidden-surface removal: Back face detection, Z-buffer method, Painter's algorithm, scan-line algorithm, BSP-trees, Area subdivision method, Ray tracing; Anti-aliasing.

UNIT V

An Introduction; Multimedia applications; Multimedia System Architecture; Evolving technologies for Multimedia; Defining objects for Multimedia systems; Multimedia Data interface standards ; Multimedia Databases; Compression & Decompression ; Data & File Format standards; Digital voice and audio; video image and animation ; Full motion video ; Storage and retrieval Technologies; Multimedia Authoring & User Interface; Hypermedia messaging; Mobile Messaging; Virtual Reality.

References:

1. Edward Angel,, Interactive Computer Graphics A Top-Down Approach with OpenGL 5th Edition, Addison-Wesley, 2008.
2. Prabat K Andleigh and Kiran Thakrar, "Multimedia Systems and Design", PHI, 2003.
3. Donald Hearn and Pauline Baker, Computer Graphics - OpenGL Version 2nd Edition, Pearson Education, 2003.
4. F.S. Hill, Jr. , Computer Graphics Using OpenGL 2nd Edition, Pearson Education, 2001.
5. Ralf Steinmetz, Klara Narstedt, Multimedia Fundamentals: Vol 1-Media Coding and Content Processing 2nd Edition, Pearson Education / PHI, 2003.

18MCASC2.6b: Cryptography and Network Security (3:1:0)

Course Outcome

- Analyze the vulnerabilities in any computing system and hence be able to design a security solution.
- Identify the security issues in the network and resolve it.
- Evaluate security mechanisms using rigorous approaches, including theoretical.
- Compare and Contrast different IEEE standards and electronic mail security.

UNIT I

Introduction-computer security concepts, attacks, security services, security mechanisms; Classical encryption techniques-symmetric cipher models, substitution techniques, transposition techniques, rotor machines

UNIT II

Symmetric ciphers-Block cipher principles; DES-Algorithm, strengths and weaknesses of DES, attacks on DES and defense, multiple encryptions; Asymmetric ciphers-Essential mathematics, public key cryptography.

UNIT III

RSA, Diffie Hellman key exchange, random number generation, Data integrity and authentication Hash functions; MAC; Digital signatures;

UNIT IV
Key management; Authentication, Web and system security, Web security; IP security; E mail security; System security-intruders, malicious software, firewalls

References

1. Cryptography and Network Security - Principles and Practice, William Stallings, PEARSON
2. Cryptography and Network Security, Atul Kahate, Tata McGraw Hill

18MCASC2.6c: System Software (3:1:0)

UNIT I
Introduction, System Software and Machine Architecture, Simplified Instructional Computer (SIC) – SIC Machine Architecture, SIC/XE Machine Architecture, SIC Programming Examples. VAX Architecture, UltraSPARC Architecture. Basic Assembler Fu nction - A Simple SIC Assembler, Assembler Algorithm and Data Structures.

UNIT II
Machine Dependent Assembler Features - Instruction Formats & Addressing Modes, Program Relocation. Machine Independent Assembler Features – Literals, Symbol-Definition Statements, Expression, Program Blocks, Control Sections and Programming Linking, Assembler Design Operations - One-Pass Assembler, Multi- Pass Assembler, Implementation Examples – MASM Assembler, SPARC Assembler

UNIT III
Basic Loader Functions - Design of an Absolute Loader, A Simple Bootstrap Loader, Machine-Dependent Loader Features – Relocation, Program Linking, Algorithm and Data Structures for a Linking Loader; Machine-Independent Loader Features – Automatic Library Search, Loader Options, Loader Design Options - Linkage Editor, Dynamic Linkage, Bootstrap Loaders, Implementation Examples – MS-DOS linker, SunOS Linker, Cray MPP linker

UNIT IV
Basic Macro Processor Functions - Macro Definitions and Expansion, Macro Processor Algorithm and Data Structures, Machine-Independent Macro Processor Features –Concatenation of Macro Parameters, Generation of Unique Labels, Conditional Macro Expansion, Keyword Macro Parameters, Macro Processor Design Options – Recursive Macro Expansion, General-Purpose Macro Processors, Macro Processing Within Language Translators, Implementation Examples - MASM Macro Processor, ANSI C Macro Processor., ELENA macro processor

UNIT V
Basic Compilers Functions- Grammars, Lexical Analysis, Syntactic Analysis, Code Generation. Machine Dependent Compiler Features- Intermediate Form of the Program, Machine dependent code Optimization. Machine Independent Compiler Features- Structured variables, Machine

Independent code Optimization. Compiler Design Options- Division into passes, Interpreters, P-code Compilers, Compiler- Compilers, SunOS C compiler, YACC Compiler-compiler.

References

1. System Software: An Introduction to Systems Programming by Leland L Beck
2. System Software by M Joseph
3. ARM System Developer's Guide: Designing and Optimizing System Software by Sloss
4. Systems and Software Quality: The next step for industrialisation by Martin Wieczorek Diederik Vos
5. System Software by Ikvinderpal Singh

18MCASC2.6d: Computer Architecture (3:1:0)

UNIT I

Basic Structure of Computers: Basic Operational Concepts, Bus Structures, Performance – Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement.

Machine Instructions and Programs: Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes, Assembly Language, Basic Input and Output Operations, Stacks and Queues, Subroutines, Additional Instructions, Encoding of Machine Instructions

UNIT II

Input/output Organization: Accessing I/O Devices, Interrupts – Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Controlling Device Requests, Exceptions, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces – PCI Bus, SCSI Bus, USB.

UNIT III

Memory System: Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories – Mapping Functions, Replacement Algorithms, Performance Considerations, Virtual Memories, Secondary Storage.

UNIT IV

Arithmetic: Numbers, Arithmetic Operations and Characters, Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed Operand Multiplication, Fast Multiplication, Integer Division, Floating-point Numbers and Operations.

UNIT V

Basic Processing Unit: Some Fundamental Concepts, Execution of a Complete Instruction, Multiple Bus Organization, Hard-wired Control, Micro programmed Control. Embedded Systems and Large Computer Systems: Examples of Embedded Systems, Processor chips for embedded applications, Simple Microcontroller. The structure of General-Purpose Multiprocessors.

References

1. Carl Hamacher, ZvonkoVranesic, SafwatZaky: Computer Organization, 5th Edition, Tata McGraw Hill.
2. Bartee C.Thomas, Digital Computer Fundamentals, McGraw-Hill International Edition
3. B. Ram, Computer Fundamentals Architecture & Organisation, New Age Pub.
4. B. Govindarajulu, "Computer Architecture and Organization: Design Principles and Applications", Second Edition, Tata McGraw-Hill.
5. M.Moris Mano, Computer Systems Architecture, 3rd Edition, PHI/Pearson.
6. Parthsarthy,Advance Computer Architecture, 2nd Edition, Thomson Learning
7. Stallings W., Computer Organisation and Architecture; Prentice Hall of India.

Continuous Assessment Scheme

C1: Test : 10 marks
Seminar/assignment : 05 marks

C2: Test : 10 marks
Seminar/assignment : 05 marks

Semester End Examination (C3)

Theory Question Paper Pattern

Time: 3 Hrs. Max . Marks: 70

- There shall be one compulsory question consisting of 5 sub questions each of 2 marks each.
- There shall be seven questions of 12 marks each.
  - Each question may have sub questions (a),(b) / (a),(b),(c)
- The student has to answer any five full questions out of remaining seven questions