

**Faculty of Science**  
**SATAVAHANA UNIVERSITY - KARIMNAGAR**  
**R19 - MCA - MASTER OF COMPUTER APPLICATIONS**  
**(Admitted 2019-20 batch)**  
Under Choice Based Credit System

**SU - M.C.A - I SEMESTER**

Paper Code	Title	Workload Per Week		Marks			Credits	Duration of the Exams.
		Theory	Practical	Internal	University	Total		
MCA 101T	Mathematical Foundations of Computer Science	4	--	20	80	100	4	3 Hrs
MCA 102T	Computer Organization and Architecture	4	--	20	80	100	4	3 Hrs
MCA 103T	Computer Programming and Data Structures	4	--	20	80	100	4	3 Hrs
MCA 104T	Computer Oriented Statistical Methods	4	--	20	80	100	4	3 Hrs
MCA 105T	Professional Communication Skills	4	--	20	80	100	4	3 Hrs
MCA 106P	Professional Communication Skills – LAB	--	4	10	40	50	2	3 Hrs
MCA 107P	Computer Programming and Data Structures – LAB	--	4	10	40	50	2	3 Hrs
MCA 107P	IT workshop – LAB	--	4	10	40	50	2	3 Hrs
<b>TOTAL</b>		<b>20</b>	<b>12</b>	<b>130</b>	<b>520</b>	<b>650</b>	<b>26</b>	

**SU M.C.A - II SEMESTER**

Paper Code	Title	Workload Per Week		Marks			Credits	Duration of the Exams.
		Theory	Practical	Internal	University	Total		
MCA 201T	Computer Networks	4	--	20	80	100	4	3 Hrs
MCA 202T	Operating Systems	4	--	20	80	100	4	3 Hrs
MCA 203T	Data Base Management Systems	4	--	20	80	100	4	3 Hrs
MCA 204T	Object Oriented Programming	4	--	20	80	100	4	3 Hrs
MCA 205T	Accountancy and Financial Management	4	--	20	80	100	4	3 Hrs
MCA 206P	Data Base Management Systems – LAB	--	4	10	40	50	2	3 Hrs
MCA 207P	Web Designing – LAB	--	4	10	40	50	2	3 Hrs
MCA 207P	Java Programming – LAB	--	4	10	40	50	2	3 Hrs
<b>TOTAL</b>		<b>20</b>	<b>12</b>	<b>130</b>	<b>520</b>	<b>650</b>	<b>26</b>	

### SU - M.C.A - III SEMESTER

Paper Code	Title	Workload Per Week		Marks			Credits	Duration of the Exams.
		Theory	Practical	Internal	University	Total		
MCA 301T	Management Information Systems	4	--	20	80	100	4	3 Hrs
MCA 302T	Operations Research	4	--	20	80	100	4	3 Hrs
MCA 303T	Software Engineering	4	--	20	80	100	4	3 Hrs
MCA 304T	Advanced Data Structures and Algorithms	4	--	20	80	100	4	3 Hrs
MCA 305T	Information Security	4	--	20	80	100	4	3 Hrs
MCA 306P	Software Engineering – LAB	--	4	10	40	50	2	3 Hrs
MCA 307P	Network Administration – LAB	--	4	10	40	50	2	3 Hrs
MCA 308P	Advanced Data Structures and Algorithms – LAB	--	4	10	40	50	2	3 Hrs
<b>TOTAL</b>		<b>20</b>	<b>12</b>	<b>130</b>	<b>520</b>	<b>650</b>	<b>26</b>	

### SU - M.C.A - IV SEMESTER

Paper Code	Title	Workload Per Week		Marks			Credits	Duration of the Exams.
		Theory	Practical	Internal	University	Total		
MCA 401T	Python Programming	4	--	20	80	100	4	3 Hrs
MCA 402T	Mobile Application Development	4	--	20	80	100	4	3 Hrs
MCA 403T	Cloud computing	4	--	20	80	100	4	3 Hrs
MCA 404T	<b>ELECTIVE – I</b> a) Automata & Compiler Design b) Computer Graphics	4	--	20	80	100	4	3 Hrs
MCA 405T	<b>ELECTIVE – II</b> a) Mobile computing b) Artificial Intelligence	4	--	20	80	100	4	3 Hrs
MCA 406P	Soft Skills – LAB	--	4	10	40	50	2	3 Hrs
MCA 407P	Python Programming – LAB	--	4	10	40	50	2	3 Hrs
MCA 408P	Mobile Application Development – LAB	--	4	10	40	50	2	3 Hrs
<b>TOTAL</b>		<b>20</b>	<b>12</b>	<b>130</b>	<b>520</b>	<b>650</b>	<b>26</b>	

### SU - M.C.A - V SEMESTER

Paper Code	Title	Workload Per Week		Marks			Credits	Duration of the Exams.
		Theory	Practical	Internal	University	Total		
MCA 501T	Data Analytics	4	--	20	80	100	4	3 Hrs
MCA 502T	Software Testing Methodologies	4	--	20	80	100	4	3 Hrs
MCA 503T	Web Programming	4	--	20	80	100	4	3 Hrs
MCA 504T	<b>ELECTIVE – I</b> a) Ethical Hacking b) Advanced Databases	4	--	20	80	100	4	3 Hrs
MCA 505T	<b>ELECTIVE – II</b> a) Internet of Things b) Image Processing	4	--	20	80	100	4	3 Hrs
MCA 506P	Data Analytics – LAB	--	4	10	40	50	2	3 Hrs
MCA 507P	Web Programming & Testing Tools – LAB	--	4	10	40	50	2	3 Hrs
MCA 508P	Project Work Review – I	--	--	50	--	50	2	--
<b>TOTAL</b>		<b>20</b>	<b>8</b>	<b>170</b>	<b>480</b>	<b>650</b>	<b>26</b>	

### SU - M.C.A - VI SEMESTER

Paper Code	Title	Workload Per Week		Marks			Credits
		Theory	Practical	Internal	University	Total	
MCA 601	Seminar	--	--	50	--	50	2
MCA 602	Project Work Review – I	--	--	100	--	100	4
MCA 603	Project Viva Voce	--	--	--	200	200	20
<b>TOTAL</b>				<b>150</b>	<b>200</b>	<b>350</b>	<b>26</b>

**SATAVAHANA UNIVERSITY KARIMNAGAR**  
**MCA I - Year I - SEM**

**MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE**

**Prerequisites:**

1. No Prerequisites
2. An understanding of Math in general is sufficient.

**Course Objectives:**

1. Introduces the elementary discrete mathematics for computer science and engineering.
2. Topics include formal logic notation, methods of proof, induction, sets, relations, graph theory, permutations and combinations, counting principles; recurrence relations and generating functions.

**Course Outcomes:**

1. Ability to understand and construct precise mathematical proofs
2. Ability to use logic and set theory to formulate precise statements
3. Ability to analyze and solve counting problems on finite and discrete structures
4. Ability to describe and manipulate sequences
5. Ability to apply graph theory in solving computing problems

**UNIT - I**

**The Foundations Logic and Proofs:** Propositional Logic, Applications of Propositional Logic, Propositional Equivalence, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference, Introduction to Proofs, Proof Methods and Strategy.

**UNIT - II**

**Basic Structures, Sets, Functions, Sequences, Sums, Matrices and Relations:** Sets, Functions , Sequences & Summations, Cardinality of Sets and Matrices Relations, Relations and Their Properties, n-ary Relations and Their Applications, Representing Relations, Closures of Relations, Equivalence Relations, Partial Orderings.

**UNIT - III**

**Algorithms, Induction and Recursion:** Algorithms, The Growth of Functions, Complexity of Algorithms.

**Induction and Recursion:** Mathematical Induction, Strong Induction and Well-Ordering, Recursive Definitions and Structural Induction, Recursive Algorithms, Program Correctness.

**UNIT - IV**

**Discrete Probability and Advanced Counting Techniques:**

An Introduction to Discrete Probability . Probability Theory, Bayes' Theorem, Expected Value and Variance.

**Advanced Counting Techniques:**

Recurrence Relations, Solving Linear Recurrence Relations, Divide-and-Conquer Algorithms and Recurrence Relations, Generating Functions, Inclusion-Exclusion, Applications of Inclusion-Exclusion.

**UNIT - V**

**Graphs:** Graphs and Graph 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.

**Trees:** Introduction to Trees, Applications of Trees, Tree Traversal, Spanning Trees, Minimum Spanning Trees.

**Textbooks:**

1. Discrete Mathematics and Its Applications with Combinatorics and Graph Theory- Kenneth H Rosen, 7<sup>th</sup> Edition, TMH.

**References:**

1. Discrete Mathematical Structures with Applications to Computer Science-J.P. Tremblay and R. Manohar, TMH,
2. Discrete Mathematics for Computer Scientists & Mathematicians: Joe L. Mott, Abraham Kandel, Theodore P. Baker, 2<sup>nd</sup> ed. , Pearson Education.
3. Discrete Mathematics- Richard Johnsonbaugh, 7<sup>th</sup> Edtn., Pearson Education.
4. Discrete Mathematics with Graph Theory- Edgar G. Goodaire, Michael M. Parmenter.
5. Discrete and Combinatorial Mathematics - an applied introduction: Ralph.P. Grimald, 5<sup>th</sup> edition , Pearson Education,.

**SATAVAHANA UNIVERSITY KARIMNAGAR**  
**MCA I - Year I - SEM**

**COMPUTER ORGANIZATION & ARCHITECTURE**

**Prerequisites** No Prerequisites

**Co-requisite:** A Course on “Digital Logic Design and Microprocessors”

**Course Objectives**

1. The purpose of the course is to introduce principles of computer organization and the basic architectural concepts.
2. It begins with basic organization, design, and programming of a simple digital computer and introduces simple register transfer language to specify various computer operations.
3. Topics include computer arithmetic, instruction set design, microprogrammed control unit, pipelining and vector processing, memory organization and I/O systems, and multiprocessors

**Course Outcomes**

1. Understand the basics of instructions sets and their impact on processor design.
2. Demonstrate an understanding of the design of the functional units of a digital computer system.
3. Evaluate cost performance and design trade-offs in designing and constructing a computer processor including memory.
4. Design a pipeline for consistent execution of instructions with minimum hazards.
5. Recognize and manipulate representations of numbers stored in digital computers

**UNIT - I**

**Digital Computers:** Introduction, Block diagram of Digital Computer, Definition of Computer Organization, Computer Design and Computer Architecture.

**Register Transfer Language and Micro operations:** Register Transfer language, Register Transfer, Bus and memory transfers, Arithmetic Micro operations, logic micro operations, shift micro operations, Arithmetic logic shift unit.

**Basic Computer Organization and Design:** Instruction codes, Computer Registers Computer instructions, Timing and Control, Instruction cycle, Memory Reference Instructions, Input – Output and Interrupt.

**UNIT - II**

**Micro Programmed Control:** Control memory, Address sequencing, micro program example, design of control unit.

**Central Processing Unit:** General Register Organization, Instruction Formats, Addressing modes, Data Transfer and Manipulation, Program Control.

**UNIT - III**

**Data Representation:** Data types, Complements, Fixed Point Representation, Floating Point Representation.

**Computer Arithmetic:** Addition and subtraction, multiplication Algorithms, Division Algorithms, Floating – point Arithmetic operations. Decimal Arithmetic unit, Decimal Arithmetic operations.

**UNIT - IV**

**Input-Output Organization:** Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt Direct memory Access.

**Memory Organization:** Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory.

**UNIT - V**

**Reduced Instruction Set Computer:** CISC Characteristics, RISC Characteristics.

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

**Multi Processors:** Characteristics of Multiprocessors, Interconnection Structures, Interprocessor arbitration, Interprocessor communication and synchronization, Cache Coherence.

**Textbooks:**

1. Computer System Architecture – M. Moris Mano, 3<sup>rd</sup> Edition, Pearson/PHI.

**References:**

1. Computer Organization – Car Hamacher, Zvonks Vranesic, SafeaZaky, 5<sup>th</sup> Edition, McGraw Hill.
2. Computer Organization and Architecture – William Stallings 6<sup>th</sup> Edition, Pearson/PHI.
3. Structured Computer Organization – Andrew S. Tanenbaum, 4<sup>th</sup> Edition PHI/Pearson.

**SATAVAHANA UNIVERSITY KARIMNAGAR**  
**MCA I - Year I - SEM**

**COMPUTER PROGRAMMING & DATA STRUCTURES**

**Prerequisites**

1. Requires analytical skills and logical reasoning.

**Course Objectives**

1. This course covers the basics of computers and program development
2. It covers various concepts of C programming language
3. It introduces searching and sorting algorithms
4. It provides an understanding of data structures such as stacks and queues.

**Course Outcomes**

1. Ability to develop C programs for computing and real life applications using basic elements like control statements, arrays, functions, pointers and strings; and data structures like stacks, queues and linked lists.
2. Ability to implement searching and sorting algorithms

**UNIT - I**

**Introduction to Computers:** Computer Systems, Computing Environments, Computer Languages, Creating and running programs, Software Development Method, Algorithms, Pseudo code, flow charts, applying the software development method.

**Introduction to C Language:** Background, Simple C programs, Identifiers, Basic data types, Variables, Constants, Input / Output, Operators. Expressions, Precedence and Associativity, Expression Evaluation, Type conversions, Bit wise operators, Statements, Simple C Programming examples.

**UNIT - II**

**Statements:** if and switch statements, Repetition statements – while, for, do-while statements, Loop examples, other statements related to looping – break, continue, go to, Simple C Programming examples.

**Designing Structured Programs:** Functions, basics, user defined functions, inter function communication, Scope, Storage classes-auto, register, static, extern, scope rules, type qualifiers, recursion- recursive functions, Preprocessor commands, example C programs

**UNIT - III**

**Arrays and Strings:** Concepts, using arrays in C, inter function communication, array applications, two – dimensional arrays, multidimensional arrays, C program examples. Concepts, C Strings, String Input / Output functions, arrays of strings, string manipulation functions, string / data conversion, C program examples.

**Pointers:** Introduction (Basic Concepts), Pointers for inter function communication, pointers to pointers, compatibility, memory allocation functions, array of pointers, programming applications, pointers to void, pointers to functions, command –line arguments.

**UNIT - IV**

**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, enumerated types, C programming examples.



**Input and Output:** Concept of a file, streams, standard input / output functions, formatted input / output functions, text files and binary files, file input / output operations, file status functions (error handling), C program examples.

#### **UNIT–V**

**Sorting and Searching:** selection sort, bubble sort, insertion sort, linear and binary search methods.

**Data Structures:** Introduction to Data Structures, abstract data types, Linear list – singly linked list implementation, insertion, deletion and searching operations on linear list, Stacks-Operations, array and linked representations of stacks, stack applications, Queues-operations, array and linked representations.

#### **Textbooks:**

1. C Programming & Data Structures, B.A.Forouzan and R.F. Gilberg,3<sup>rd</sup> Edition, Cengage Learning.
2. Problem Solving and Program Design in C, J.R. Hanly and E.B. Koffman, 5<sup>th</sup> Edition, Pearson Education.
3. The C Programming Language, B.W. Kernighan and Dennis M.Ritchie, PHI/Pearson Education

#### **References:**

1. C for Engineers and Scientists, H.Cheng, Mc.Graw-Hill International Edition
2. Data Structures using C – A.M.Tanenbaum, Y.Langsam, and M.J. Augenstein, Pearson Education / PHI
3. C Programming & Data Structures, P. Dey, M Ghosh R Thereja, Oxford University Press

**SATAVAHANA UNIVERSITY KARIMNAGAR**  
**MCA I - Year I - SEM**

**COMPUTER ORIENTED STATISTICAL METHODS**

**Prerequisites** No pre requisites, Foundation course

**Objectives:** The aim of the course is to understand

1. The theory of Probability, and probability distributions of single and multiple random variables
2. The sampling theory and testing of hypothesis and making inferences
3. The regression and correlation

**Outcomes:** At the end of the course student is able to

1. Apply the concepts of probability and distributions to some case studies
2. Correlate the material of one unit to the material in other units
3. Resolve the potential misconceptions and hazards in each topic of study.

**UNIT - I**

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

**Random Variables and Probability Distributions:** Concept of a Random Variable, Discrete Probability Distributions, Continuous Probability Distributions, Statistical Independence, 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.

**Discrete Probability Distributions:** Introduction and Motivation, Binomial and Multinomial Distributions, Hypergeometric Distribution, Negative Binomial and Geometric Distributions, Poisson distribution.

**UNIT - III**

**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, Beta Distribution, Lognormal Distribution.

**Fundamental Sampling Distributions:** Random Sampling, Some Important Statistics, Sampling Distributions, Sampling Distribution of Means and the Central Limit Theorem, Sampling Distribution of  $S^2$ ,  $t$ -Distribution, F-Distribution.

**UNIT - IV**

**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, Estimating the Variance, Estimating a Proportion. **Two Samples:** Estimating the Difference between Two Means, Paired Observations, Estimating the Difference between Two Proportions, Estimating the Ratio of Two Variances, Maximum Likelihood Estimation.

**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, Two Samples: Tests on Two Means, Choice of Sample Size for Testing Means, Graphical Methods for Comparing Means, One Sample: Test on a Single Proportion, Two Samples: Tests on Two Proportions, One- and Two-Sample Tests Concerning Variances.

## **UNIT - V**

### **Linear Regression and Correlation**

Introduction to Linear Regression, The Simple Linear Regression Model, Least Squares and the Fitted Model, Properties of the Least Squares Estimators, Inferences Concerning the Regression Coefficients, Prediction, Choice of a Regression Model, Analysis-of-Variance Approach, Test for Linearity of Regression: Data with Repeated Observations, Correlation.

#### **Textbook:**

1. Probability & Statistics For Engineers & Scientists, by Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Keying Ye. 9<sup>th</sup> Ed. Pearson Pub.

#### **References:**

1. Fundamentals Of Probability And Statistics For Engineers, T.T. Soong, *John Wiley & Sons, Ltd, 2004.*
2. Fundamentals of Mathematical statistics, khanna publications, S C Gupta and V K Kapoor.
3. Probability and statistics for Engineers and scientists, academic press, Sheldon M Ross.

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA I - Year I - SEM****PROFESSIONAL COMMUNICATION SKILLS**

**Prerequisites** ---- NIL ----

**Course Objectives**

1. To learn the four language skills - Listening, Speaking, Reading and Writing; critical thinking skills to students.
2. To enable students comprehend the concept of communication.
3. To help students cultivate the habit of Reading and develop their critical reading skills.

**Course Outcomes**

1. Ability to convert the conceptual understanding of communication into every day practice.
2. Ability to communicate their ideas relevantly and coherently in professional writing.

**UNIT - I****Introduction:**

Basics of Communication - Principles of Communication - Types of Communication – Stages of Communication – Verbal and Non-verbal Communication – Channels of Communication – Barriers to Effective Communication – Formal and Informal Expressions in Various Situations.

**UNIT - II****Reading & Study Skills:**

Reading Comprehension – Reading Strategies - Skimming and Scanning- Intensive and Extensive Reading– Unknown Passage for Comprehension - Critical Reading of Short Stories – Study Skills – Note Making – Summarizing – Articles and Prepositions – Synonyms and Antonyms

**UNIT - III****Writing Skills:**

Difference between Spoken and Written Communication- Features of Effective Writing - Formation of a Sentence – SVOs and SVOC patterns – Types of sentences- Common errors in Writing - Writing coherent sentences using connectives and conjunctions- Written Presentation Skills – Tenses – Concord – Question Tags - Practice Exercises - One Word Substitutes – Words Often Confused and Misspelt.

**UNIT - IV****Professional Writing:**

Letter writing – Types, Parts and Styles of Formal Letters – Language to be used in Formal Letters – Letters of Enquiry, Complaint, and Apology with Replies – Letter of Application -Resume – E-mail – Active and Passive Voice.

**UNIT - V****Report Writing:**

Types of Reports – Formats of Reports – Memo Format – Letter Format and Manuscript Format- Parts of Technical Report – Informational, Analytical and Project Reports – Idioms and Phrases.

**References:**

1. Meenakshi Raman & Sangeetha Sharma. 2012. *Technical Communication*. New Delhi: OUP
2. Rizvi, M. A. 2005. *Effective Technical Communication*. New Delhi: Tata McGraw Hill

3. Sanjay Kumar & Pushp Latha. 2012. *Communication Skills*. New Delhi: OUP
4. Er. A. K. Jain, Dr. Pravin S. R. Bhatia & Dr. A. M. Sheikh. 2013. *Professional Communication Skills*. S. Chand Publishers. New Delhi.
5. Farhathullah, T.M. 2009. *English for Business Communication*. Bangalore: Prism Publishers
6. Bikram K Das. 2011. *Functional Grammar and Spoken and Written Communication in English*. Kolkata: Orient Blackswan
7. Kiranmai Dutt, P *et al.* 2011. *A Course in Communication Skills*. New Delhi: CUP India
8. Krishnaswamy, N. 2000. *Modern English – A Book of Grammar, Vocabulary and Usage*. Macmillan India Pvt. Ltd
9. Ramachandran, K K. *et al.* 2007. *Business Communication*. New Delhi: Macmillan
10. Taylor, Ken. 2011. *50 ways to improve your Business English*. Hyderabad: Orient Blackswan

**SATAVAHANA UNIVERSITY KARIMNAGAR**

**MCA I - Year I - SEM**

**PROFESSIONAL COMMUNICATION SKILLS LAB**

**Prerequisites ----NIL---**

**Course Objectives:**

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

**Course Outcomes:**

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

**Syllabus: Professional Communication Skills Lab shall have two parts:**

- a. **Computer Assisted Language Learning (CALL) Lab**
- b. **Interactive Communication Skills (ICS) Lab**

The following course content is prescribed for the **Professional Communication Skills Lab**

**Exercise – I**

**CALL Lab:** Introduction to Phonetics – Speech Sounds – Vowels and Consonants

**ICS Lab:** Ice-Breaking activity and JAM session

Articles, Prepositions, Word formation- Prefixes & Suffixes, Synonyms & Antonyms

**Exercise – II**

**CALL Lab:** Structure of Syllables - Past Tense Marker and Plural Marker – Weak Forms and Strong Forms - Consonant Clusters.

**ICS Lab:** Situational Dialogues – Role-Play- Expressions in Various Situations – Self-introduction and Introducing Others – Greetings – Apologies – Requests – Social and Professional Etiquette - Telephone Etiquette.

Concord (Subject in agreement with verb) and Words often misspelt- confused/misused

**Exercise - III**

**CALL Lab:** Minimal Pairs- Word accent and Stress Shifts- Listening Comprehension.

**ICS Lab:** Descriptions- Narrations- Giving Directions and guidelines.

Sequence of Tenses, Question Tags and One word substitutes.

**Exercise – IV**

**CALL Lab:** Intonation and Common errors in Pronunciation.

**ICS Lab:** Extempore- Public Speaking  
Active and Passive Voice, –Common Errors in English, Idioms and Phrases

### Exercise – V

**CALL Lab:** Neutralization of Mother Tongue Influence and Conversation Practice

**ICS Lab:** Information Transfer- Oral Presentation Skills

Reading Comprehension and Job Application with Resume preparation.

### Minimum Requirement of infrastructural facilities for PCS Lab:

#### 1. Computer Assisted Language Learning (CALL) Lab:

**The Computer aided Language Lab** for 40 students with 40 systems, one master console, LAN facility and English language software for self- study by learners.

#### System Requirement (Hardware component):

Computer network with Lan with minimum 60 multimedia systems with the following specifications:

- i) P – IV Processor
  - a) Speed – 2.8 GHZ
  - b) RAM – 512 MB Minimum
  - c) Hard Disk – 80 GB
- ii) Headphones of High quality

#### 2. Interactive Communication Skills (ICS) Lab :

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

#### Suggested Software:

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

#### SUGGESTED READING:

1. Rama Krishna Rao, A. *et al. English Language Communication Skills – A Reader cum Lab Manual Course Content and Practice.* Chennai: Anuradha Publishers
2. Suresh Kumar, E. & Sreehari, P. 2009. *A Handbook for English Language Laboratories.* New Delhi: Foundation
3. *Speaking English Effectively* 2<sup>nd</sup> Edition by Krishna Mohan and N. P. Singh, 2011. Macmillan Publishers India Ltd. Delhi.
4. Sasi Kumar, V & Dhamija, P.V. *How to Prepare for Group Discussion and Interviews.* Tata McGraw Hill

5. Spoken English: A Manual of Speech and Phonetics by R. K. Bansal & J. B. Harrison. 2013. Orient Blackswan. Hyderabad.
6. *English Pronunciation in Use. (Elementary, Intermediate & Advance)*. Cambridge: CUP
7. Chris Redston, Gillie Cunningham, Jan Bell. *Face to Face* (2<sup>nd</sup> Edition). Cambridge University Press
8. Nambiar, K.C. 2011. *Speaking Accurately. A Course in International Communication*. New Delhi : Foundation
9. Soundararaj, Francis. 2012. *Basics of Communication in English*. New Delhi: Macmillan
10. A textbook of English Phonetics for Indian Students by T. Balasubramanian (Macmillan)

#### **DISTRIBUTION AND WEIGHTAGE OF MARKS**

##### ***Professional Communication Skills Lab Practical Examination:***

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

\* \* \* \* \*



**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA I - Year I - SEM****COMPUTER PROGRAMMING & DATA STRUCTURES LAB****Prerequisites:**

1. Requires analytical skills and logical reasoning

**Course Objectives:**

1. It covers various concepts of C programming language
2. It introduces searching and sorting algorithms
3. It provides an understanding of data structures such as stacks and queues.

**Course Outcomes:**

1. Develop C programs for computing and real life applications using basic elements like control statements, arrays, functions, pointers and strings, and data structures like stacks, queues and linked lists.
2. Implement searching and sorting algorithms

**Week 1:**

1. Write a C program to find the sum of individual digits of a positive integer.
2. 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.
3. Write a C program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.
4. Write a C program to find the roots of a quadratic equation.

**Week 2:**

5. Write a C program to find the factorial of a given integer.
6. Write a C program to find the GCD (greatest common divisor) of two given integers.
7. Write a C program to solve Towers of Hanoi problem.
8. Write a C program, which takes two integer operands and one operator from the user, performs the operation and then prints the result. (Consider the operators +, -, \*, /, % and use Switch Statement)

**Week 3:**

9. Write a C program to find both the largest and smallest number in a list of integers.
10. Write a C program that uses functions to perform the following:
  - i) Addition of Two Matrices
  - ii) Multiplication of Two Matrices

**Week 4:**

11. Write a C program that uses functions to perform the following operations:
  - i) To insert a sub-string in to a given main string from a given position.
  - ii) To delete n Characters from a given position in a given string.
12. Write a C program to determine if the given string is a palindrome or not
13. Write a C program that displays the position or index in the string S where the string T begins, or - 1 if S doesn't contain T.
14. Write a C program to count the lines, words and characters in a given text.

**Week 5:**

15. Write a C program to generate Pascal's triangle.

16. Write a C program to construct a pyramid of numbers.
17. Write a C program to read in two numbers, x and n, and then compute the sum of this geometric progression:

$$1+x+x^2+x^3+\dots+x^n$$

For example: if n is 3 and x is 5, then the program computes 1+5+25+125.

Print x, n, the sum

Perform error checking. For example, the formula does not make sense for negative exponents – if n is less than 0. Have your program print an error message if  $n < 0$ , then go back and read in the next pair of numbers without computing the sum. Are any values of x also illegal? If so, test for them too.

**Week 6:**

18. 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.
19. Write a C program to convert a Roman numeral to its decimal equivalent.

**Week 7:**

20. Write a C program that uses functions to perform the following operations:

- i) Reading a complex number
- ii) Writing a complex number
- iii) Addition of two complex numbers
- iv) Multiplication of two complex numbers

(Note: represent complex number using a structure.)

**Week 8:**

21. i) Write a C program which copies one file to another.  
ii) 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.)
22. i) Write a C program to display the contents of a file.  
ii) Write a C program to merge two files into a third file (i.e., the contents of the first file followed by those of the second are put in the third file)

**Week 9:**

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

**Week 10:**

24. Write C programs that implement stack (its operations) using  
i) Arrays                      ii) Pointers
25. Write C programs that implement Queue (its operations) using  
i) Arrays                      ii) Pointers

**Week 11:**

26. Write a C program that implements the following sorting methods to sort a given list of integers in ascending order  
i) Bubble sort    ii) Selection sort

**Week 12:**

27. 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

**Textbooks:**

1. C Programming & Data Structures, B.A.Forouzan and R.F. Gilberg, 3<sup>rd</sup> Edition, Cengage Learning.
2. Problem Solving and Program Design in C, J.R. Hanly and E.B. Koffman, 5<sup>th</sup> Edition, Pearson Education.
3. The C Programming Language, B.W. Kernighan and Dennis M.Ritchie, PHI/Pearson Education

**References:**

1. C for Engineers and Scientists, H.Cheng, Mc.Graw-Hill International Edition
2. Data Structures using C – A.M.Tanenbaum, Y.Langsam, and M.J. Augenstein, Pearson Education / PHI
3. C Programming & Data Structures, P. Dey, M Ghosh R Thereja, Oxford University Press

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA I - Year I - SEM  
IT WORKSHOP LAB****Course Objectives**

1. The IT Workshop is a training lab course to get training on PC Hardware, Internet & World Wide Web, and Productivity tools for documentation, Spreadsheet computations, and Presentation.
2. To introduce to a personal computer and its basic peripherals, the process of assembling a personal computer, installation of system software like MS Windows, Linux and the required device drivers, hardware and software level troubleshooting process.
3. To introduce connecting the PC on to the internet from home and workplace and effectively usage of the internet, Usage of web browsers, email, newsgroups and discussion forums. To get knowledge in awareness of cyber hygiene, i.e., protecting the personal computer from getting infected with the viruses, worms and other cyber attacks.
4. To introduce the usage of Productivity tools in crafting professional word documents, excel spreadsheets and power point presentations using open office tools and LaTeX.

**Course Outcomes**

1. Apply knowledge for computer assembling and software installation.
2. Ability how to solve the trouble shooting problems.
3. Apply the tools for preparation of PPT, Documentation and budget sheet etc.

**PC Hardware:** The students should work on working PC to disassemble and assemble to working condition and install operating system like Linux or any other on the same PC. Students are suggested to work similar tasks in the Laptop scenario wherever possible.

**Problem 1:** Every student should identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram of the CPU along with the configuration of each peripheral and submit to your instructor. Every student should disassemble and assemble the PC back to working condition.

**Problem 2:** Every student should individually install operating system like Linux or MS windows on the personal computer. The system should be configured as dual boot with both windows and Linux.

**Problem 3:** Hardware Troubleshooting: Students have to be given a PC which does not boot due to improper assembly or defective peripherals. They should identify the problem and fix it to get the computer back to working condition.

**Problem 4:** Software Troubleshooting: Students have to be given a malfunctioning CPU due to system software problems. They should identify the problem and fix it to get the computer back to working condition.

**Internet & World Wide Web.**

**Problem 5:** Orientation & Connectivity Boot Camp: Students should get connected to their Local Area Network and access the Internet. In the process they configure the TCP/IP setting. Finally students should demonstrate how to access the websites and email.

**Problem 6:** Web Browsers, Surfing the Web: Students customize their web browsers with the LAN proxy settings, bookmarks, search toolbars and pop up blockers. Also, plug-ins like Macromedia Flash and JRE for applets should be configured.

**Problem 7:** Search Engines & Netiquette: Students should know what search engines are and how to use the search engines. Usage of search engines like Google, Yahoo, ask.com and others should be demonstrated by student.

**Problem 8:** Cyber Hygiene: Students should learn about viruses on the internet and install antivirus software. Student should learn to customize the browsers to block pop ups, block active x downloads to avoid viruses and/or worms.

**Problem 9:** Develop home page: Student should learn to develop his/her home page using HTML consisting of his/her photo, name, address and education details as a table and his/her skill set as a list.

**Productivity tools: LaTeX and Word**

**Word Orientation:** An overview of LaTeX and Microsoft (MS) office / equivalent (FOSS) tool word should be learned: Importance of LaTeX and MS office / equivalent (FOSS) tool Word as word Processors, Details of the three tasks and features that should be covered in each, using LaTeX and word – Accessing, overview of toolbars, saving files, Using help and resources, rulers, format painter.

**Problem 10: Using LaTeX and Word** to create project certificate. Features to be covered:-Formatting Fonts in word, Drop Cap in word, Applying Text effects, Using Character Spacing, Borders and Colors, Inserting Header and Footer, Using Date and Time option in both LaTeX and Word.

**Problem 11: Creating project abstract** Features to be covered:-Formatting Styles, Inserting table, Bullets and Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check, Track Changes.

**Problem 12: Creating a Newsletter:** Features to be covered:- Table of Content, Newspaper columns, Images from files and clipart, Drawing toolbar and Word Art, Formatting Images, Textboxes, Paragraphs in word.

**Problem 13 - Spreadsheet Orientation:** Accessing, overview of toolbars, saving spreadsheet files, Using help and resources. **Creating a Scheduler:-** Gridlines, Format Cells, Summation, auto fill, Formatting Text

**Problem 14: Calculating GPA -** .Features to be covered:- Cell Referencing, Formulae in spreadsheet – average, std. deviation, Charts, Renaming and Inserting worksheets, Hyper linking, Count function, Sorting, Conditional formatting.

**Problem 15: Creating Power Point:** Student should work on basic power point utilities and tools in Latex and Ms Office/equivalent (FOSS) which help them create basic power point presentation. PPT Orientation, Slide Layouts, Inserting Text, Formatting Text, Bullets and Numbering, Auto Shapes, Lines and Arrows, Hyperlinks, Inserting Images, Tables and Charts

**Textbooks:**

1. Introduction to Information Technology, ITL Education Solutions limited, Pearson Education.
2. LaTeX Companion – Leslie Lamport, PHI/Pearson.
3. Comdex Information Technology course tool kit Vikas Gupta, WILEY Dreamtech
4. IT Essentials PC Hardware and Software Companion Guide by David Anfinson and Ken Quamme. 3<sup>rd</sup> Edition – CISCO Press, Pearson Education.
5. PC Hardware and A+ Handbook – Kate J. Chase PHI (Microsoft)

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA I - Year II - SEM****COMPUTER NETWORKS****Prerequisites**

1. A course on "Programming for problem solving"
2. A course on "Data Structures"

**Course Objectives**

1. The objective of the course is to equip the students with a general overview of the concepts and fundamentals of computer networks.
2. Familiarize the students with the standard models for the layered approach to communication between machines in a network and the protocols of the various layers.

**Course Outcomes**

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

**UNIT - I**

Network hardware, Network software, OSI, TCP/IP Reference models, Example Networks: ARPANET, Internet.

**Physical Layer:** Guided Transmission media: twisted pairs, coaxial cable, fiber optics, Wireless transmission.

**UNIT - II**

**Data link layer:** Design issues, framing, Error detection and correction.

Elementary data link protocols: simplex protocol, A simplex stop and wait protocol for an error-free channel, A simplex stop and wait protocol for noisy channel.

Sliding Window protocols: A one-bit sliding window protocol, A protocol using Go-Back-N, A protocol using Selective Repeat, Example data link protocols.

Medium Access sub layer: The channel allocation problem, Multiple access protocols: ALOHA, Carrier sense multiple access protocols, collision free protocols. Wireless LANs, Data link layer switching.

**UNIT - III**

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

**UNIT - IV**

**Transport Layer:** Transport Services, Elements of Transport protocols, Connection management, TCP and UDP protocols.

**UNIT - V**

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

**Textbook:**

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

**References:**

2. An Engineering Approach to Computer Networks-S.Keshav,2<sup>nd</sup> Edition,Pearson Education
3. Data Communications and Networking – Behrouz A. Forouzan. 3<sup>rd</sup> Edition TMH.

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA I - Year II - SEM  
OPERATING SYSTEMS****Prerequisites:**

1. A course on "Computer Programming and Data Structures"
2. A course on "Computer Organization and Architecture"

**Course Objectives:**

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

**Course Outcomes:**

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

**UNIT - I**

**Operating System Introduction:** Structures - Simple Batch, Multiprogrammed, Time-shared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, System components, Operating System services, System Calls.

**UNIT - II**

**Process and CPU Scheduling:** Process concepts and scheduling, Operations on processes, Cooperating Processes, Threads, and Interposes Communication, Scheduling Criteria, Scheduling Algorithms, Multiple -Processor Scheduling.

**System call interface for process management:** fork, exit, wait, waitpid, exec

**UNIT - III**

**Deadlocks:** System Model, Deadlocks Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock.

**Process Management and Synchronization:** The Critical Section Problem, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Critical Regions, Monitors.

**Interprocess Communication Mechanisms:** IPC between processes on a single computer system, IPC between processes on different systems, using pipes, FIFOs, message queues, shared memory.

**UNIT - IV**

**Memory Management and Virtual Memory:** Logical versus Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging, Demand Paging, Page Replacement, Page Replacement Algorithms.

**UNIT - V**

**File System Interface and Operations:** Access methods, Directory Structure, Protection, File System Structure, Allocation methods, Free-space Management. Usage of open, create, read, write, close, lseek, stat, ioctl, system calls.



**Textbooks:**

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

**References;**

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

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA I - Year II - SEM****DATABASE MANAGEMENT SYSTEMS****Prerequisite:**

1. A course on "Data Structures".

**Course Objectives:**

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

**Course Outcomes:**

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

**UNIT - I**

**Database System Applications:** A Historical Perspective, File Systems versus a DBMS, the Data Model, Levels of Abstraction in a DBMS, Data Independence, Structure of a DBMS

**Introduction to Database Design:** Database Design and ER Diagrams, Entities, Attributes, and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design With the ER Model

**UNIT - II**

**Introduction to the Relational Model:** Integrity constraint over relations, enforcing integrity constraints, querying relational data, logical data base design, introduction to views, destroying/altering tables and views.

Relational Algebra, Tuple relational Calculus, Domain relational calculus.

**UNIT - III**

**SQL: Queries, Constraints, Triggers:** form of basic SQL query, UNION, INTERSECT, and EXCEPT, Nested Queries, aggregation operators, NULL values, complex integrity constraints in SQL, triggers and active data bases.

**Schema refinement:** Problems caused by redundancy, decompositions, problems related to decomposition, reasoning about functional dependencies, FIRST, SECOND, THIRD normal forms, BCNF, lossless join decomposition, multi-valued dependencies, FOURTH normal form, FIFTH normal form.

**UNIT - IV**

Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for serializability, Lock Based Protocols, Timestamp Based Protocols, Validation- Based Protocols, Multiple Granularity, Recovery and Atomicity, Log-Based Recovery, Recovery with Concurrent Transactions.

**UNIT - V**

Data on External Storage, File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes, Index data Structures, Hash Based Indexing, Tree base Indexing, Comparison of File

Organizations, Indexes and Performance Tuning, Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM), B+ Trees: A Dynamic Index Structure.

**Text Books:**

1. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, 3rd Edition, Tata Mc Graw Hill
2. Database System Concepts, Silberschatz, Korth, 5<sup>th</sup> edition, Mc Graw hill.

**References:**

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

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA I - Year II - SEM****OBJECT ORIENTED PROGRAMMING****Prerequisites:**

1. A course on "Computer Programming & Data Structures"

**Course Objectives:**

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

**Course Outcomes:**

1. Develop applications for a range of problems using object-oriented programming techniques
2. Design simple Graphical User Interface applications

**UNIT - I**

**Object oriented thinking and Java Basics:** Need for oop paradigm, summary of oop concepts, History of Java, Java buzzwords, data types, variables, scope and life time of variables, arrays, operators, expressions, control statements, type conversion and casting, simple java program, concepts of classes, objects, constructors, methods, access control, this keyword, using final with variables, garbage collection, overloading methods and constructors, recursion, nested and inner classes, exploring string class.

**UNIT - II**

**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 and methods, polymorphism- method overriding, abstract classes, the Object class.

**Packages:** Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages.

**UNIT - III**

**Interfaces:** differences between classes and interfaces, defining an interface, implementing interface, applying interfaces, variables in interface and extending interfaces.

**Exception handling:** Concepts of exception handling, benefits of exception handling, Termination or resumptive models, exception hierarchy, usage of try, catch, throw, throws and finally, built in exceptions, creating own exception sub classes.

String handling, Exploring java.util

**UNIT - IV**

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

**Swing:** Introduction, limitations of AWT, MVC architecture, components, containers, exploring swing- JApplet, JFrame and JComponent, Icons and Labels, text fields, buttons – The JButton class, Check boxes, Radio buttons, Combo boxes, Tabbed Panes, Scroll Panes, Trees, and Tables.

Handling menus, graphics, layout manager – layout manager types – border, grid, flow, card and grid bag.

#### **UNIT - V**

**Applets:** Concepts of Applets, differences between applets and applications, life cycle of an applet, types of applets, creating applets, passing parameters to applets.

**Multithreading:** Differences between multi threading and multitasking, thread life cycle, creating threads, thread priorities, synchronizing threads, interthread communication, thread groups.

#### **TEXTBOOKS:**

1. Java the complete reference, Herbert schildt, 7<sup>th</sup> editon, TMH.
2. Understanding OOP with Java, updated edition, T. Budd, pearson education.

#### **REFERENCES:**

1. An Introduction to programming and OO design using Java, J.Nino and F.A. Hosch, John wiley & sons.
2. An Introduction to OOP, T. Budd, 3<sup>rd</sup> edition, pearson education.
3. Introduction to Java programming, Y. Daniel Liang, pearson education.
4. An introduction to Java programming and object oriented application development, R.A. Johnson- Thomson.
5. Core Java 2, Vol 1, Fundamentals, Cay.S.Horstmann and Gary
6. Cornell, 8<sup>th</sup> Edition, Pearson Education.
7. 6. Core Java 2, Vol 2, Advanced Features, Cay.S.Horstmann and Gary Cornell, 8<sup>th</sup> Edition, Pearson Education
7. Object Oriented Programming with Java, R.Buyya,S.T.Selvi,X.Chu,TMH.
8. Java and Object Orientation, an introduction, John Hunt, 2<sup>nd</sup> edition, Springer.
9. Maurach's Beginning Java2 JDK 5 , SPD.
10. Programming and Problem Solving with Java, JM Slack, B S Publications.

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA I - Year II – SEM****ACCOUNTANCY AND FINANCIAL MANAGEMENT****Course Objectives:**

1. To learn Financial Management and Accounting
2. To learn different types of costing

**Course Outcomes:**

1. Able to prepare balance sheets of budget.
2. Get the skill to manage finances of a firm/company

**UNIT - I**

**Introduction to Accounting:** Principles, concepts, conventions, double entry system of accounting, introduction of basic books of accounts ledgers.

**Preparation of trial balance:** Final accounts - company final accounts.

**UNIT - II**

**Financial Management:** meaning and scope, role, objectives of time value of money - over vitalization - under capitalization - profit maximization - wealth maximization - EPS maximization.

**Ratio Analysis:** advantages - limitations - Fund flow analysis - meaning, importance, preparation and interpretation of Funds flow and cash flow statements-statement.

**UNIT - III**

**Costing:** nature and importance and basic principles. 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, limitations and importance of cost - volume, profit analysis.

**UNIT - IV**

**Standard costing and budgeting:** nature, scope and computation and analysis - materials variance, labor variance and sales variance - budgeting - cash budget, sales budget - flexible Budgets, master budgets.

**UNIT - V**

**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.

**TEXTBOOKS:**

1. Van Horne, James, C: Financial Management and Policy, 12<sup>th</sup> Edition, Pearson Education.
2. Financial Accounting, S.N.Maheshwari, Sultan Chand Company.
3. Financial Management, S.N.Maheshwari, Sultan Chand Company

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA I - Year II - SEM****DATABASE MANAGEMENT SYSTEMS LAB****Co-requisites**

1. Co-requisite of course "Database Management Systems"

**Course Objectives**

1. Introduce ER data model, database design and normalization
2. Learn SQL basics for data definition and data manipulation

**Course Outcomes**

1. Design database schema for a given application and apply normalization
2. Acquire skills in using SQL commands for data definition and data manipulation.
3. Develop solutions for database applications using procedures, cursors and triggers

**List of Experiments:-**

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

**Textbooks:**

1. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, 3<sup>rd</sup> Edition Tata Mc Graw Hill
2. Database System Concepts, Silberschatz, Korth,, 5<sup>th</sup> edition, Mc Graw hill.

**References:**

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

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA I - Year II - SEM****WEB DESIGNING LAB****Prerequisites:**

1. A Course on "Computer Programming and Data Structures"
2. A Course on "Objected Oriented Programming through Java"

**Co-requisites:**

1. A course on "Web Technologies"

**Course Objectives:**

1. To provide hands-on experience on web technologies
2. To develop client-server application using web technologies
3. To introduce server side programming with Java servlets and JSP

**Course Outcomes:**

1. Design and develop interactive and dynamic web applications using HTML, CSS, JavaScript and XML
2. Apply client-server principles to develop scalable and enterprise web applications.

**List of Experiments:**

1. Develop static pages (using Only HTML) of an online book store. The pages should resemble: [www.amazon.com](http://www.amazon.com). The website should consist the following pages.
  - a) Home page
  - b) Registration and user Login
  - c) User Profile Page
  - d) Books catalog
  - e) Shopping Cart
  - f) Payment By credit card
  - g) Order Conformation
2. Validate the Registration, user login, user profile and payment by credit card pages using JavaScript.
3. Create and save an XML document on the server, which contains 10 users information. Write a program, which takes User Id as an input and returns the user details by taking the user information from the XML document.
4. Bean Assignments
  - a. Create a JavaBean which gives the exchange value of INR (Indian Rupees) into equivalent American/Canadian/Australian Dollar value.
  - b. Create a simple Bean with a label - which is the *count* of number of clicks. Then create a BeanInfo class, such that only the "*count*" property is visible in the Property Window.
  - c. Create two Beans-      a)KeyPad.                      b)DisplayPad.  
After that integrate the two Beans to make it work as a Calculator.
  - d. Create two Beans: Traffic Light(Implemented as a Label with only three background colours - Red,Green,Yellow) and Automobile (Implemented as a TextBox which states its state/movement). The state of the Automobile should depend on the following Light Transition Table.



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

5. Install TOMCAT web server. Convert the static web pages of assignments 2 into dynamic web pages using servlets and cookies. Hint: Users information (user id, password, credit card number) would be stored in web.xml. Each user should have a separate Shopping Cart.
6. Redo the previous task using JSP by converting the static web pages of assignments 2 into dynamic web pages. Create a database with user information and books information. The books catalogue should be dynamically loaded from the database. Follow the MVC architecture while doing the website.

**Textbooks:**

1. WEB TECHNOLOGIES: A Computer Science Perspective, Jeffrey C. Jackson, Pearson Education

**References:**

1. Deitel H.M. and Deitel P.J., "Internet and World Wide Web How to program", 4<sup>th</sup> Edition Pearson International, 2012.
2. J2EE: The complete Reference By James Keogh, McGraw-Hill
3. Bai and Ekedhi, The Web Warrior Guide to Web Programming, Thomson
4. Paul Dietel and Harvey Deitel,"Java How to Program", 8<sup>th</sup> Edition, Prentice Hall of India
5. Web technologies, Black Book, Dreamtech press.
6. Gopalan N.P. and Akilandeswari J., "Web Technology", Prentice Hall of India

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA I - Year II - SEM****JAVA PROGRAMMING LAB****Prerequisites**

1. A course on "Computer Programming & Data Structures"

**Co-requisite:**

1. A Course on "Object-Oriented Programming Through Java"

**Course Objectives:**

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

**Course Outcomes:**

1. Develop applications for a range of problems using object-oriented programming techniques
2. Design simple Graphical User Interface applications

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

- 1) Write a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method printArea () that prints the area of the given shape.
- 2) Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the exception in a message dialog box. [Use JOption Pane –Input dialog, Message dialog]
- 3) Write a Java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.
- 4) Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with "Stop" or "Ready" or "Go" should appear above the buttons in selected color. Initially, there is no message shown.
- 5) Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -, \*, % operations. Add a text field to display the result. Handle any possible exceptions like divide by zero
- 6) a) Develop an applet in Java that displays a simple message.

- b) Develop an applet in Java that receives an integer in one text field, and computes its factorial value and returns it in another text field, when the button named "Compute" is clicked.
- 7) Write a Java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired (Using Adapter classes).
- 8) Write a Java program that handles all keyboard events and shows the event name at the center of the window when a mouse event is fired (Use Adapter classes).

**Textbooks:**

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

**References:**

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

**SATAVAHANA UNIVERSITY KARIMNAGAR**

**MCA II - Year I - SEM**

**MANAGEMENT INFORMATION SYSTEMS**

**Prerequisites**----NIL----

**Course Objectives:**

1. To introduce Information Systems Models
2. Topics include Types of Information Systems, ERP Modules etc.,

**Course Outcomes:**

1. To understand different Types of Information Systems
2. To gain good knowledge of ERP Modules and ERP Implementation and Maintenance.

**UNIT - I**

**Introduction to IS Models:** Nolan Stage Hypothesis, IS Strategic Grid, Wards Model, Earl's Multiple Methodology, Critical Success Factors, Soft Systems Methodology, Socio-Technical Systems Approach (Mumford), System Develop Life Cycle, Prototype and End User Computing, Application Packages, Outsourcing, Deciding Combination of Methods.

**UNIT - II**

**Types of Information Systems:** Transactions Processing System, Knowledge Work Systems, Office Automation System, Management Information System, Decision Support System, Expert System, Strategic Information System. IS Security, Control and Audit - System Vulnerability and Abuse, business value of security and control, Need for Security, Methods of minimizing risks, IS Audit, ensuring system quality.

**UNIT - III**

**Induction to ERP:** Overview of ERP, MRP, MRPII and Evolution of ERP, Integrated Management Systems, Reasons for the growth of ERP, Business Modeling, Integrated Data Model, Foundations of IS in Business, Obstacles of applying IT. Advantages and limitations of ERP.

**UNIT - IV**

**ERP Modules:** Finance, Accounting Systems, Manufacturing and Production Systems, Sales and Distribution Systems, , Human Resource Systems, Plant Maintenance System, Materials Management System, Quality Management System, ERP System Options and Selection, ERP proposal Evaluation.

**UNIT - V**

**ERP Implementation and Maintenance:** Implementation Strategy Options, Features of Successful ERP Implementation, Strategies to Attain Success, User Training, Maintaining ERP & IS. Case Studies.

**Textbooks:**

1. Information systems for modern management, 3<sup>rd</sup> Edition by R.G Murdick, J.E Ross And J. R clagget, PHI-1994.

**References:**

1. C Laudon and Jane P.Laudon, et al: Management Information Systems, Pearson Education, 2009.
2. Alexis Leon, ERP (Demystified), 5/E, Tata McGraw-Hill, 2009.
3. David L Olson, Managerial Issues of Enterprise Resource Planning Systems, McGraw Hill, International Edition-2009.

4. Vaman, ERP in Practice, Tata McGraw-Hill , 2009
5. Gordon B. Davis & Margrethe H.Olson: Management Information Systems, Tata McGraw-Hill , 2009.
6. W S Jawadekar: Management Information Systems, Tata McGraw-Hill , New Delhi, 2009
7. James A. Obrein: Management Information Systems, Tata McGraw-Hill , 2008
8. Gerald V.Post, David L Anderson: Management Information Systems, Irvin McGraw Hill, 2009.

**SATAVAHANA UNIVERSITY KARIMNAGAR**

**MCA II - Year I - SEM**

**OPERATIONS RESEARCH**

**Prerequisites:** A course on “Mathematics”

**Course Objectives:**

1. This course explains various optimization problems and the techniques to address those problems.
2. To study Linear Programming, dynamic programming and optimization Techniques.
3. To understand the theory of games.

**Course Outcomes:**

1. Gain the knowledge of optimization techniques
2. Get the skill to apply Optimization techniques to address the real time problems .

**UNIT - I**

Development – Definition– Characteristics and Phases – Types of models – Operations Research models – applications.

**Allocation:** Linear Programming Problem - Formulation – Graphical solution – Simplex method – Artificial variables techniques: Two–phase method, Big-M method; Duality Principle.

**UNIT - II**

**Sequencing:** n jobs through two machines – n jobs through three machines – Job shop sequencing – two jobs through ‘m’ machines.

**Replacement:** Introduction – Replacement of items that deteriorate with time – when money value is not counted and counted

**UNIT - III**

**Transportation Problem:** Formulation – Optimal solution, unbalanced transportation problem – Degeneracy.

**Assignment problem:** Formulation – Optimal solution - Variants of Assignment Problem, Travelling Salesman problem.

**UNIT - IV**

**Theory Of Games:** Introduction –Terminology– Solution of games with saddle points and without saddle points- 2 x 2 games –m x 2 & 2 x n games - graphical method – m x n games -dominance principle.

**Waiting Lines:** Introduction – Terminology-Single Channel – Poisson arrivals and Exponential Service times – with infinite population and finite population models.

**UNIT - V**

**Inventory:** Introduction – Single item, Deterministic models – Types - Purchase inventory models with one price break and multiple price breaks

**Dynamic Programming:**

Introduction – Terminology- Bellman’s Principle of Optimality – Applications of dynamic programming- shortest path problem – linear programming problem.

**Textbooks:**

1. Operation Research /J.K.Sharma/MacMilan.

2. Introduction to O.R /Taha/PHI
3. Operations Research/S.D.Sharma

**References:**

1. Operations Research: Methods and Problems / Maurice Saseini, ArthurYaspan and Lawrence Friedman
2. Operations Research /A.M.Natarajan, P.Balasubramaniam, A. Tamilarasi/Pearson Education.
3. Operations Research / Wagner/ PHI Publications.
4. Introduction to O.R/Hillier &Libermann (TMH).

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA II - Year I - SEM****SOFTWARE ENGINEERING****Course Objectives:**

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

**Course Outcomes:**

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

**UNIT - I**

**Introduction to Software Engineering:** The evolving role of software, changing nature of software, software myths.

**A Generic view of process:** Software engineering- a layered technology, a process framework, the capability maturity model integration (CMMI), process patterns, process assessment, personal and team process models.

**Process models:** The waterfall model, incremental process models, evolutionary process models, the unified process.

**UNIT - II**

**Software Requirements:** Functional and non-functional requirements, user requirements, system requirements, interface specification, the software requirements document.

**Requirements engineering process:** Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management.

**System models:** Context models, behavioral models, data models, object models, structured methods.

**UNIT - III**

**Design Engineering:** Design process and design quality, design concepts, the design model.

**Creating an architectural design:** software architecture, data design, architectural styles and patterns, architectural design, conceptual model of UML, basic structural modeling, class diagrams, sequence diagrams, collaboration diagrams, use case diagrams, component diagrams.

**UNIT - IV**

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

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

**UNIT - V**

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



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

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

**Textbooks:**

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6<sup>th</sup> edition, Mc GrawHill International Edition.
2. Software Engineering- Sommerville, 7<sup>th</sup> edition, Pearson Education.
3. The unified modeling language user guide Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education.

**References:**

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

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA II - Year I - SEM****ADVANCED DATA STRUCTURES & ALGORITHMS****Prerequisites:**

A course on "Computer Programming & Data Structures"

**Course Objectives:**

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

**Course Outcomes:**

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

**UNIT - I**

**Introduction:** Algorithm, Psuedo code for expressing algorithms, Performance Analysis-Space complexity, Time complexity, Asymptotic Notations, Substitution method,

**Divide and conquer:** General method , applications-Binary search, Quick sort, Merge sort, Strassen's matrix multiplication

**UNIT - II**

**Disjoint Sets:** Disjoint set operations - union and find algorithms

**Greedy method:** General method, applications-Job sequencing with deadlines, knapsack problem, Minimum cost spanning trees

**Priority Queues** – Definition, Realizing a Priority Queue using Heaps, operations of priority queue

**UNIT - III**

**Dynamic Programming:** General method, applications- Optimal binary search trees, 0/1 knapsack problem, Travelling sales person problem, Reliability design.

**Backtracking:** General method, applications-n-queen's problem, sum of subsets problem.

**UNIT - IV**

**Review of basic data structures:** The list, Stack, Queues, Linear list representation, skip list representation, operations - insertion, deletion and searching.

**Hash table representation:** hash functions, collision resolution-separate chaining, open addressing-linear probing, quadratic probing, double hashing, rehashing, extendible hashing, comparison of hashing and skip lists.

**Searching:** Linear Search and Binary Search

**Sortings:** Bubble sort, Selection sort, Insertion sort, Radix sort, Heap sort

**UNIT - V**

**Trees:** Definition, Types of trees, Binary Trees, Binary Tree Traversal Methods, Binary search tree, AVL tree and B-Tree operations-insertion,deletion and searching

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

**Textbooks:**

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

**References:**

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

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA II - Year I - SEM****INFORMATION SECURITY****Prerequisites**

1. A Course on "Computer Networks and a course on Mathematics

**Course Objectives:**

1. To understand the fundamentals of Cryptography
2. To understand various key distribution and management schemes
3. To understand how to deploy encryption techniques to secure data in transit across data networks
4. To apply algorithms used for secure transactions in real world applications

**Course Outcomes:**

1. Demonstrate the knowledge of cryptography, network security concepts and applications.
2. Ability to apply security principles in system design.
3. Ability to identify and investigate vulnerabilities and security threats and mechanisms to counter them.

**UNIT - I**

Security Attacks (Interruption, Interception, Modification and Fabrication), Security Services (Confidentiality, Authentication, Integrity, Non-repudiation, access Control and Availability) and Mechanisms, A model for Internetwork security.

Classical Encryption Techniques, DES, Strength of DES, Differential and Linear Cryptanalysis, Block Cipher Design Principles and Modes of operation, Blowfish, Placement of Encryption Function, Traffic Confidentiality, key Distribution, Random Number Generation.

**UNIT - II**

Public key Cryptography Principles, RSA algorithm, Key Management, Diffie-Hellman Key Exchange, Elliptic Curve Cryptography.

Message authentication and Hash Functions, Authentication Requirements and Functions, Message Authentication, Hash Functions and MACs Hash and MAC Algorithms SHA-512, HMAC.

**UNIT - III**

Digital Signatures, Authentication Protocols, Digital signature Standard, Authentication Applications, Kerberos, X.509 Directory Authentication Service.

Email Security: Pretty Good Privacy (PGP) and S/MIME.

**UNIT - IV**

**IP Security:** Overview, IP Security Architecture, Authentication Header, Encapsulating Security Payload, Combining Security Associations and Key Management.

**Web Security:** Web Security Requirements, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET).

**UNIT - V**

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

**Text Books:**

1. Cryptography and Network Security (principles and approaches) by William Stallings, 4<sup>th</sup> Edition Pearson Education.

**References:**

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

**SATAVAHANA UNIVERSITY KARIMNAGAR**  
**MCA II - Year I - SEM**  
**SOFTWARE ENGINEERING LAB**

**Prerequisites**

A course on "Programming for Problem Solving"

**Co-requisite:**

A Course on "Software Engineering"

**Course Objectives:**

1. To have hands on experience in developing a software project by using various software engineering principles and methods in each of the phases of software development.

**Course Outcomes:**

1. Ability to translate end-user requirements into system and software requirements
2. Ability to generate a high level design of the system from the software requirements
3. Will have experience and/or awareness of testing problems and will be able to develop a simple testing report

**LIST OF EXPERIMENTS:**

**Do the following 8 exercises for any two projects given in the list of sample projects or any other projects:**

- 1) Development of problem statement.
- 2) Preparation of Software Requirement Specification Document, Design Documents and Testing Phase related documents.
- 3) Preparation of Software Configuration Management and Risk Management related documents.
- 4) Study and usage of any Design phase CASE tool
- 5) Performing the Design by using any Design phase CASE tools.
- 6) Develop test cases for unit testing and integration testing
- 7) Develop test cases for various white box and black box testing techniques.

**Sample Projects:**

1. Passport automation System
2. Book Bank
3. Online Exam Registration
4. Stock Maintenance System
5. Online course reservation system
6. E-ticketing
7. Software Personnel Management System
8. Credit Card Processing
9. E-book management System.
10. Recruitment system

**Textbooks**

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6<sup>th</sup> edition, Mc GrawHill International Edition.
2. Software Engineering- Sommerville, 7<sup>th</sup> edition, Pearson Education.
3. The unified modeling language user guide Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education.

**SATAVAHANA UNIVERSITY KARIMNAGAR**

**MCA II - Year I - SEM**

**NETWORK ADMINISTRATION LAB**

**Prerequisites**

1. Computer Networks

**Course Objectives:**

1. Practical exposure to students in network installation, administration and maintenance

**Course Outcomes:**

1. Student understands Linux commands for configuring system
2. Student acquires requisite skills in network installation, administration maintenance

**Shell Programming:**

1. Working with shell commands related to file permissions, processes, networking and backup.
2. Write a shell script to generate a multiplication table
3. Write a shell script that copies multiple files to a directory
4. Write a shell script which counts the number of lines and words present in a given file
5. Write a shell script that displays the list of files in the given directory
6. Write a shell script that adds, subtracts, multiplies and divides the given two numbers. There are two division options: one returns the quotient and the other returns remainder. The script requires 3 arguments: The operation to be used and two integer numbers. The options are add(-a), subtract (-s), multiply (-m) and remainder (-r).
7. Write a shell script to reverse the rows and columns of a matrix

**System and network experiments:**

1. Demonstration of server construction
2. Create the "LVM" with the name "development" by using 21PE's from the volume group "engineering". Consider the PE size as "8MB". Mount it on /mnt/secret with filesystem ext3.
3. Create a group named "sysadmin"  
A user curly and larry should belongs to "sysadmin" group as a secondary group . A user node should not have access to interactive shell and he should not be a member of "sysadmin" group. passwd for all user created should be "jenny".
4. Create the Directory "/home/manager" with the following characteristics. Group ownership of "/home/manager" should go to "manager" group. The directory should be have full permission for all members off "manager" group but not to any other users accept "root". Files created under "/home/manager" should get the same group ownership is set to the "manager" group
5. The user sarah must configure a cron job that runs daily at 14:23. and executes "/bin/echo "hyer" and deny the user max for creating cronjob
6. Resize the lvm "/dev/vgsrv/home" so that after reboot size should be in between 90MB to 120MB
7. Configure your system as "NTP" client for "instructor.example.com".
8. Note the following. instructor.example.com(192.168.0.254) "Nfs exports" /home/guests to your system where "x" is your station ip. Ldapuser's home directory is instructor.example.com:/home/guests/ldapuserX. Ldapuser's home directory should be automounted locally beneath at /home/guests/ldapuserX. While login with any of the ldapuser then only home directory should accesible from your system that ldapuserX.
9. Copy the file /etc/fstab to /var/tmp and configure the "ACL" as mention following. The file /var/tmp/fstab is owned by the "root". The file /var/tmp/fstab belongs to the group "root"

The file `/var/tmp/fstab` should not be executable by other's. The user "sarah" should be able to read and write to the file. The user "natasha" can neither read nor write to the file. Other users (future and current) should be able to read `/var/tmp/fstab`.

10: Create the user "dax" with uid 4223

11: Extend the SWAP space with "250" MB don't remove the existing swap.

12: Configure FTP access from your system. Clients within the remote.test should not have anonymous FTP access to your system.

13: locate the files of owner "dax" and copy to the directory `/root/found` directory

14: List all lines which have string "enter" from `/usr/share/dict/words` file and copy the lines in `/root/word.found`.

**Textbooks:**

1. Operating systems Internals and Design Principles, 8<sup>th</sup> edition, William Stallings, Pearson.
2. Unix the ultimate guide, sumithabha Das, TMH
3. Microsoft® Windows Server® 2008 Administration, STEVE SEGUIS, Mc Graw Hill.
4. Red Hat Enterprise Linux 6 Administration, Sander van Vugt, John Wiley & Sons.

**References:**

1. The Complete Reference Linux, Richard Petersen, Mc Graw Hill.
2. Maurice J. Bach, "Design of UNIX Operating System", PHI.
3. Linux system Administration, Tom Adelstein & Bill Lubanovic, O'Reilly.



**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA II - Year I - SEM****ADVANCED DATA STRUCTURES & ALGORITHMS LAB****Prerequisites**

1. A course on “Computer Programming & Data Structures” and “ Advanced Data Base Engineering”.

**Course Objectives**

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

**Course Outcomes**

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

**List of Experiments:**

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

**Text Books:**

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

**References:**

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

**SATAVAHANA UNIVERSITY KARIMNAGAR**

**MCA II - Year II - SEM**

**PYTHON PROGRAMMING**

**Course Objectives:** This course will enable students to

1. Learn Syntax and Semantics and create Functions in Python.
2. Handle Strings and Files in Python.
3. Understand Lists, Dictionaries and Regular expressions in Python.
4. Implement Object Oriented Programming concepts in Python.
5. Build Web Services and introduction to Network and Database Programming in Python.

**Course Outcomes:** The students should be able to:

1. Examine Python syntax and semantics and be fluent in the use of Python flow control and functions.
2. Demonstrate proficiency in handling Strings and File Systems.
3. Create, run and manipulate Python Programs using core data structures like Lists, Dictionaries and use Regular Expressions.
4. Interpret the concepts of Object-Oriented Programming as used in Python.
5. Implement exemplary applications related to Network Programming, Web Services and Databases in Python.

**UNIT - I**

Python Basics, Objects- Python Objects, Standard Types, Other Built-in Types, Internal Types, Standard Type Operators, Standard Type Built-in Functions, Categorizing the Standard Types, Unsupported Types

Numbers - Introduction to Numbers, Integers, Floating Point Real Numbers, Complex Numbers, Operators, Built-in Functions, Related Modules

Sequences - Strings, Lists, and Tuples, Mapping and Set Types

**UNIT - II**

FILES: File Objects, File Built-in Function [ open() ], File Built-in Methods, File Built-in Attributes, Standard Files, Command-line Arguments, File System, File Execution, Persistent Storage Modules, Related Modules

Exceptions: Exceptions in Python, Detecting and Handling Exceptions, Context Management, \*Exceptions as Strings, Raising Exceptions, Assertions, Standard Exceptions, \*Creating Exceptions, Why Exceptions (Now)?, Why Exceptions at All?, Exceptions and the sys Module, Related Modules

Modules: Modules and Files, Namespaces, Importing Modules, Importing Module Attributes, Module Built-in Functions, Packages, Other Features of Modules

**UNIT - III**

Regular Expressions: Introduction, Special Symbols and Characters, Res and Python

Multithreaded Programming: Introduction, Threads and Processes, Python, Threads, and the Global Interpreter Lock, Thread Module, Threading Module, Related Modules

**UNIT - IV**

GUI Programming: Introduction, Tkinter and Python Programming, Brief Tour of Other GUIs, Related Modules and Other GUIs

WEB Programming: Introduction, Web Surfing with Python, Creating Simple Web Clients, Advanced Web Clients, CGI-Helping Servers Process Client Data, Building CGI Application

Advanced CGI, Web (HTTP) Servers

**UNIT – V**

Database Programming:

Introduction, Python Database Application Programmer's Interface (DB-API), Object Relational Managers (ORMs), Related Modules

**Textbook:**

1. Core Python Programming, Wesley J. Chun, Second Edition, Pearson.

**SATAVAHANA UNIVERSITY KARIMNAGAR**  
**MCA II - Year II - SEM**  
**MOBILE APPLICATION DEVELOPMENT**

**Prerequisites:**

1. Acquaintance with JAVA programming
2. A Course on DBMS

**Course Objectives**

1. To demonstrate their understanding of the fundamentals of Android operating systems
2. To improve their skills of using Android software development tools
3. To demonstrate their ability to develop software with reasonable complexity on mobile platform
4. To demonstrate their ability to deploy software to mobile devices
5. To demonstrate their ability to debug programs running on mobile devices

**Course Outcomes:**

1. Student understands the working of Android OS Practically.
2. Student will be able to develop Android user interfaces
3. Student will be able to develop, deploy and maintain the Android Applications.

**UNIT - I**

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

**UNIT - II**

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

**UNIT - III**

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

**UNIT - IV**

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

**UNIT - V**

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

**Textbooks:**

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

**References:**

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

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA II - Year II - SEM****CLOUD COMPUTING****Prerequisites:**

1. A course on "Computer Networks"
2. A course on "Operating Systems"
3. A course on "Distributed Systems"

**Course Objectives:**

1. This course provides an insight into cloud computing
2. Topics covered include- distributed system models, different cloud service models, service oriented architectures, cloud programming and software environments, resource management.

**Course Outcomes:**

1. Ability to understand various service delivery models of a cloud computing architecture.
2. Ability to understand the ways in which the cloud can be programmed and deployed.
3. Understanding cloud service providers.

**UNIT - I**

**Computing Paradigms:** High-Performance Computing, Parallel Computing, Distributed Computing, Cluster Computing, Grid Computing, Cloud Computing, Bio computing, Mobile Computing, Quantum Computing, Optical Computing, Nano computing.

**UNIT - II**

**Cloud Computing Fundamentals:** Motivation for Cloud Computing, The Need for Cloud Computing, Defining Cloud Computing, Definition of Cloud computing, Cloud Computing Is a Service, Cloud Computing Is a Platform, Principles of Cloud computing, Five Essential Characteristics, Four Cloud Deployment Models

**UNIT - III**

**Cloud Computing Architecture and Management:** Cloud architecture, Layer, Anatomy of the Cloud, Network Connectivity in Cloud Computing, Applications, on the Cloud, Managing the Cloud, Managing the Cloud Infrastructure Managing the Cloud application, Migrating Application to Cloud, Phases of Cloud Migration Approaches for Cloud Migration.

**UNIT - IV**

**Cloud Service Models:** Infrastructure as a Service, Characteristics of IaaS. Suitability of IaaS, Pros and Cons of IaaS, Summary of IaaS Providers, Platform as a Service, Characteristics of PaaS, Suitability of PaaS, Pros and Cons of PaaS, Summary of PaaS Providers, Software as a Service, Characteristics of SaaS, Suitability of SaaS, Pros and Cons of SaaS, Summary of SaaS Providers, Other Cloud Service Models.

**UNIT - V**

**Cloud Service Providers:** EMC, EMC IT, Captiva Cloud Toolkit, Google, Cloud Platform, Cloud Storage, Google Cloud Connect, Google Cloud Print, Google App Engine, Amazon Web Services, Amazon Elastic Compute Cloud, Amazon Simple Storage Service, Amazon Simple Queue ,service, Microsoft, Windows Azure, Microsoft Assessment and Planning Toolkit, SharePoint, IBM, Cloud Models, IBM Smart Cloud, SAP Labs, SAP HANA Cloud Platform, Virtualization Services Provided by

SAP, Sales force, Sales Cloud, Service Cloud: Knowledge as a Service, Rack space, VMware, Manjra soft, Aneka Platform

**Textbooks:**

1. Essentials of cloud Computing : K.Chandrasekhran , CRC press, 2014

**References:**

1. Cloud Computing: Principles and Paradigms by Rajkumar Buyya, James Broberg and Andrzej M. Goscinski, Wiley, 2011.
2. Distributed and Cloud Computing , Kai Hwang, Geoffery C.Fox, Jack J.Dongarra, Elsevier, 2012.
3. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, Subra Kumaraswamy, Shahed Latif, O'Reilly, SPD, rp2011.



**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA II - Year II - SEM****MOBILE COMPUTING (Professional Elective - I)****Prerequisites:**

1. A course on "Computer Networks"

**Course Objectives:**

1. To make the student understand the concept of mobile computing paradigm, its novel applications and limitations.
2. To understand the typical mobile networking infrastructure through a popular GSM protocol
3. To understand the issues and solutions of various layers of mobile networks, namely MAC layer, Network Layer & Transport Layer
4. To understand the database issues in mobile environments & data delivery models.
5. To understand the ad hoc networks and related concepts.
6. To understand the platforms and protocols used in the mobile environment.

**Course Outcomes:**

1. Able to think and develop new mobile application.
2. Able to take any new technical issue related to this new paradigm and come up with a solution(s).
3. Able to develop new ad hoc network applications and/or algorithms/protocols.
4. Able to understand & develop any existing or new protocol related to the mobile environment

**UNIT - I****Introduction**

Mobile Communications, Mobile Computing – Paradigm, Promises/Novel Applications and Impediments and Architecture; Mobile and Handheld Devices, Limitations of Mobile and Handheld Devices.

GSM – Services, System Architecture, Radio Interfaces, Protocols, Localization, Calling, Handover, Security, New Data Services, GPRS.

**UNIT - II****(Wireless) Medium Access Control (MAC)**

Motivation for a specialized MAC (Hidden and exposed terminals), SDMA, FDMA, TDMA, CDMA, Wireless LAN/(IEEE 802.11)

**Mobile Network Layer**

IP and Mobile IP Network Layers, Packet Delivery and Handover Management, Location Management, Registration, Tunneling and Encapsulation, Route Optimization, DHCP.

**UNIT - III****Mobile Transport Layer**

Conventional TCP/IP Protocols, Indirect TCP, Snooping TCP, Mobile TCP, Other Transport Layer Protocols for Mobile Networks.

**Database Issues**

Database Hoarding & Caching Techniques, Client-Server Computing & Adaptation, Transactional Models.

**UNIT - IV**

**Data Dissemination and Synchronization**

Communications Asymmetry, Classification of Data Delivery Mechanisms, Data Dissemination, Broadcast Models, Selective Tuning and Indexing Methods.

**UNIT - V**

**Mobile Ad hoc Networks (MANETs)**

Introduction, Applications & Challenges of a MANET, Routing, Classification of Routing Algorithms, Algorithms such as DSR, AODV, DSDV, Mobile Agents, Service Discovery.

**Textbooks:**

1. "Mobile Communications", Jochen Schiller, 2<sup>nd</sup> Edition Addison-Wesley, 2009.
2. "Mobile Computing", Raj Kamal, Oxford University Press, 2007, ISBN: 0195686772

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA II - Year II - SEM****AUTOMATA & COMPILER DESIGN (Professional Elective - I)****Prerequisites:**

1. A course on "Computer Organization and architecture"
2. A course on "Computer Programming and Data Structures"

**Course Objectives:**

1. To provide introduction to some of the central ideas of theoretical computer science from the perspective of formal languages.
2. To introduce the fundamental concepts of formal languages, grammars and automata theory.
3. Classify machines by their power to recognize languages.
4. Employ finite state machines to solve problems in computing.
5. To understand deterministic and non-deterministic machines.
6. Introduce the major concepts of language translation and compiler design and impart the knowledge of practical skills necessary for constructing a compiler.
7. Topics include phases of compiler, parsing, code optimization techniques, intermediate code generation, code generation.

**Course Outcomes:**

1. Able to understand the concept of abstract machines and their power to recognize the languages.
2. Able to employ finite state machines for modeling and solving computing problems.
3. Able to design context free grammars for formal languages.
4. Demonstrate the ability to design a compiler given a set of language features.
5. Demonstrate the knowledge of patterns, tokens & regular expressions for lexical analysis.
6. Design and implement LL and LR parsers
7. Design algorithms to do code optimization in order to improve the performance of a program in terms of space and time complexity.
8. Design algorithms to generate machine code.

**UNIT - I**

**Introduction to Finite Automata:** Structural Representations, Automata and Complexity, the Central Concepts of Automata Theory – Alphabets, Strings, Languages, Problems.

**Nondeterministic Finite Automata:** Formal Definition, an application, Text Search, Finite Automata with Epsilon-Transitions.

**Regular Expressions:** Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions, Conversion of Finite Automata to Regular Expressions.

**UNIT - II**

**Context-Free Grammars:** Definition of Context-Free Grammars, Derivations Using a Grammar, Leftmost and Rightmost Derivations, the Language of a Grammar, Sentential Forms, Parse Trees, Applications of Context-Free Grammars, Ambiguity in Grammars and Languages.

**Push Down Automata:** Definition of the Pushdown Automaton, the Languages of a PDA, Equivalence of PDA's and CFG's, Acceptance by final state, Acceptance by empty stack, Deterministic Pushdown Automata. From CFG to PDA.

**UNIT - III**

**Turing Machines:** Introduction to Turing Machine, Formal Description, Instantaneous description, The language of a Turing machine

**Introduction to compiler:** The structure of a compiler, the science of building a compiler, programming language basics

**UNIT - IV**

**Syntax Analysis:** Introduction, Context-Free Grammars, Writing a Grammar, Top-Down Parsing, Bottom-Up Parsing, Introduction to LR Parsing: Simple LR, More Powerful LR Parsers, Using Ambiguous Grammars and Parser Generators.

**UNIT - V**

**Intermediate-Code Generation:** Variants of Syntax Trees, Three-Address Code, Types and Declarations, Type Checking, Control Flow, Switch-Statements, Intermediate Code for Procedures.

**Code Generation:** Issues in the Design of a Code Generator, The Target Language, Addresses in the Target Code, A Simple Code Generator

**Machine-Independent Optimization:** The Principal Sources of Optimization

**Textbooks:**

1. Introduction to Automata Theory, Languages, and Computation, John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, 3<sup>rd</sup> Edition, Pearson Education.
2. Compilers: Principles, Techniques and Tools, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, 2<sup>nd</sup> Edition,

**References:**

1. Theory of Computer Science – Automata languages and computation, Mishra and Chandrashekar, 2<sup>nd</sup> edition, PHI.
2. Introduction to Languages and The Theory of Computation, John C Martin, TMH.
3. Introduction to Computer Theory, Daniel I.A. Cohen, John Wiley.
4. A Text book on Automata Theory, P. K. Srimani, Nasir S. F. B, Cambridge University Press.
5. Introduction to the Theory of Computation, Michael Sipser, 3<sup>rd</sup> edition, Cengage Learning.
6. Introduction to Formal languages Automata Theory and Computation Kamala Krithivasan, Rama R, Pearson.

**SATAVAHANA UNIVERSITY KARIMNAGAR**  
**MCA II - Year II - SEM**

**ARTIFICIAL INTELLIGENCE (Professional Elective - I)**

**Prerequisites:**

1. A course on "Computer Programming and Data Structures"
2. A course on "Advanced Data Structures"
3. A course on "Design and Analysis of Algorithms"
4. A course on "Mathematical Foundations of Computer Science"
5. Some background in linear algebra, data structures and algorithms, and probability will all be helpful

**Course Objectives:**

1. To learn the distinction between optimal reasoning Vs. human like reasoning
2. To understand the concepts of state space representation, exhaustive search, heuristic search together with the time and space complexities.
3. To learn different knowledge representation techniques.
4. To understand the applications of AI, namely game playing, theorem proving, and machine learning.

**Course Outcomes:**

1. Ability to formulate an efficient problem space for a problem expressed in natural language.
2. Select a search algorithm for a problem and estimate its time and space complexities.
3. Possess the skill for representing knowledge using the appropriate technique for a given problem.
4. Possess the ability to apply AI techniques to solve problems of game playing, and machine learning.

**UNIT - I**

**Problem Solving by Search-I**

Introduction to AI, Intelligent Agents

**Problem Solving by Search –II:**

Problem-Solving Agents, Searching for Solutions, Uninformed Search Strategies: Breadth-first search, Uniform cost search, Depth-first search, Iterative deepening Depth-first search, Bidirectional search, Informed (Heuristic) Search Strategies: Greedy best-first search, A\* search, Heuristic Functions, Beyond Classical Search: Hill-climbing search, Simulated annealing search, Local Search in Continuous Spaces, Searching with Non-Deterministic Actions, Searching with Partial Observations, Online Search Agents and Unknown Environment .

**UNIT - II**

**Problem Solving by Search-II and Propositional Logic**

**Adversarial Search:**

Games, Optimal Decisions in Games, Alpha–Beta Pruning, Imperfect Real-Time Decisions.

**Constraint Satisfaction Problems :**

Defining Constraint Satisfaction Problems, Constraint Propagation, Backtracking Search for CSPs, Local Search for CSPs, The Structure of Problems.

**Propositional Logic:**

Knowledge-Based Agents, The Wumpus World, Logic, Propositional Logic, Propositional Theorem Proving: Inference and proofs , Proof by resolution, Horn clauses and definite clauses, Forward and backward chaining , Effective Propositional Model Checking , Agents Based on Propositional Logic.

**UNIT - III****Logic and Knowledge Representation****First-Order Logic:**

Representation, Syntax and Semantics of First-Order Logic, Using First-Order Logic, Knowledge Engineering in First-Order Logic.

**Inference in First-Order Logic :**

Propositional vs. First-Order Inference, Unification and Lifting, Forward Chaining, Backward Chaining, Resolution.

**Knowledge Representation :**

Ontological Engineering, Categories and Objects, Events. Mental Events and Mental Objects, Reasoning Systems for Categories, Reasoning with Default Information.

**UNIT - IV****Planning****Classical Planning:**

Definition of Classical Planning, Algorithms for Planning with State-Space Search, Planning Graphs, other Classical Planning Approaches, Analysis of Planning approaches.

**Planning and Acting in the Real World :**

Time, Schedules, and Resources, Hierarchical Planning, Planning and Acting in Nondeterministic Domains, Multi agent Planning,.

**UNIT - V Uncertain knowledge and Learning****Uncertainty :**

Acting under Uncertainty, Basic Probability Notation, Inference Using Full Joint Distributions, Independence, Bayes' Rule and Its Use,

**Probabilistic Reasoning:**

Representing Knowledge in an Uncertain Domain, The Semantics of Bayesian Networks, Efficient Representation of Conditional Distributions, Approximate Inference in Bayesian Networks, Relational and First-Order Probability, Other Approaches to Uncertain Reasoning ; Dempster-Shafer theory.

**Learning:**

Forms of Learning, Supervised Learning, Learning Decision Trees. Knowledge in Learning : Logical Formulation of Learning, Knowledge in Learning, Explanation-Based Learning, Learning Using Relevance Information, Inductive Logic Programming.

**Textbooks:**

1. Artificial Intelligence A Modern Approach, Stuart Russell and Peter Norvig, 3<sup>rd</sup> Edition, Pearson Education.

**References:**

1. Artificial Intelligence , 3<sup>rd</sup> Edn. , E.Rich and K.Knight (TMH)
2. Artificial Intelligence, 3<sup>rd</sup> Edn., Patrick Henny Winston, Pearson Education.
3. Artificial Intelligence, Shivani Goel, Pearson Education.
4. Artificial Intelligence and Expert systems – Patterson, Pearson Education.

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA II - Year II - SEM****COMPUTER GRAPHICS (Professional Elective - I)****Prerequisites:**

1. Familiarity with the theory and use of coordinate geometry and of linear algebra such as matrix multiplication.
2. A course on "Computer Programming and Data Structures"

**Course Objectives:**

1. The aim of this course is to provide an introduction of fundamental concepts and theory of computer graphics.
2. Topics covered include graphics systems and input devices; geometric representations and 2D/3D transformations; viewing and projections; illumination and color models; animation; rendering and implementation; visible surface detection;

**Course Outcomes:**

1. Acquire familiarity with the relevant mathematics of computer graphics.
2. Be able to design basic graphics application programs, including animation
3. Be able to design applications that display graphic images to given specifications

**UNIT - I**

**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 (Bresenham's and DDA Algorithm), mid-point circle and ellipse algorithms

**Polygon Filling:** Scan-line algorithm, boundary-fill and flood-fill algorithms

**UNIT - II**

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

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

**UNIT - III**

**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.

**UNIT - IV**

**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**

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

**Visible surface detection methods:** Classification, back-face detection, depth-buffer, BSP-tree methods and area sub-division methods

### **Text Books:**

1. "Computer Graphics *C version*", Donald Hearn and M.Pauline Baker, Pearson Education
2. "Computer Graphics Principles & practice", Foley, VanDam, Feiner and Hughes, 2<sup>nd</sup> edition in C, Pearson Education.
3. Computer Graphics, Steven Harrington, TMH

### **References:**

1. Procedural elements for Computer Graphics, David F Rogers, Tata Mc Graw hill, 2<sup>nd</sup> edition.
2. Principles of Interactive Computer Graphics", Neuman and Sproul, TMH.
3. Principles of Computer Graphics, Shalini Govil, Pai, 2005, Springer.



3. Marketing, Paul Baines, Chris Fill, Kelly Page, Piyush Sinha, Asian Edition, Oxford University Press, 2015.
4. Marketing Management , Arun Kumar & N. Meenakshi, Vikas, 2012
5. Marketing Management, Rajan Saxena, 3<sup>rd</sup>e, Tata Mc Graw Hill, 2012.
6. Cases in Marketing Management, Kenneth E Clow, Donald Baack, Sage South Asia edition, 2012.

## SATAVAHANA UNIVERSITY KARIMNAGAR

### MCA II - Year II - SEM

#### SOFT SKILLS LAB

#### Course Objectives:

1. To improve the fluency of students in English
2. To facilitate learning through interaction
3. To illustrate the role of skills in real-life situations with case studies, role plays etc.
4. To train students in group dynamics, body language and various other activities which boost their confidence levels and help in their overall personality development
5. To encourage students develop behavioral skills and personal management skills
6. To impart training for empowerment, thereby preparing students to become successful professionals

#### Course Outcomes:

1. Developed critical acumen and creative ability besides making them industry- ready.
2. Appropriate use of English language while clearly articulating ideas.
3. Developing insights into Language and enrich the professional competence of the students.
4. Enable students to meet challenges in job and career advancement.

#### INTRODUCTION

Definition and Introduction to Soft Skills – Hard Skills vs Soft Skills – Significance of Soft/Life/Self Skills – Self and SWOT Analysis *and*

##### 1. Exercises on Productivity Development

- Effective/ Assertive Communication Skills (Activity based)
- Time Management (Case Study)
- Creativity & Critical Thinking (Case Study)
- Decision Making and Problem Solving (Case Study)
- Stress Management (Case Study)

##### 2. Exercises on Personality Development Skills

- Self-esteem (Case Study)
- Positive Thinking (Case Study)
- Emotional Intelligence (Case Study)
- Team building and Leadership Skills (Case Study)
- Conflict Management (Case Study)

##### 3. Exercises on Presentation Skills

- Netiquette
- Importance of Oral Presentation – Defining Purpose- Analyzing the audience- Planning Outline and Preparing the Presentation- Individual & Group Presentation- Graphical Organizers- Tools and Multi-media Visuals
- One Minute Presentations (Warming up)
- PPT on Project Work- Understanding the Nuances of Delivery- Body Language – Closing and Handling Questions – Rubrics for Individual Evaluation (Practice Sessions)

##### 4. Exercises on Professional Etiquette and Communication

- Role-Play and Simulation- Introducing oneself and others, Greetings, Apologies, Requests, Agreement & Disagreement....etc.
- Telephone Etiquette
- Active Listening

- Group Discussions (Case study)- Group Discussion as a part of Selection Procedure- Checklist of GDs
- Analysis of Selected Interviews (Objectives of Interview)
- Mock-Interviews (Practice Sessions)
- Job Application and Preparing Resume
- Process Writing (Technical Vocabulary) – Writing a Project Report- Assignments

#### 5. Exercises on Ethics and Values

Introduction — Types of Values - Personal, Social and Cultural Values - Importance of Values in Various Contexts

- Significance of Modern and Professional Etiquette – Etiquette (Formal and Informal Situations with Examples)
- Attitude, Good Manners and Work Culture (Live Examples)
- Social Skills - Dealing with the Challenged (Live Examples)
- Professional Responsibility – Adaptability (Live Examples)
- Corporate Expectations

☞ Note: Hand-outs are to be prepared and given to students.

☞ Training plan will be integrated in the syllabus.

☞ Topics mentioned in the syllabus are activity-based.

#### SUGGESTED SOFTWARE:

☞ The following software from ‘[train2success.com](http://train2success.com)’

- Preparing for being Interviewed
- Positive Thinking
- Interviewing Skills
- Telephone Skills
- Time Management
- Team Building
- Decision making

#### SUGGESTED READING:

1. Alex, K. 2012. *Soft Skills*. S. Chand Publishers
2. *Management Shapers*. 2011. Collection of 28 Books by different Authors. Universities Press.
3. Sherfield, Robert M. 2005. *et al Cornerstone: Developing Soft Skills*. Pearson
4. Suresh Kumar, E; Sreehari, P. & Savithri, J. 2011. *Communication Skills and Soft Skills- An Integrated Approach*. New Delhi: Pearson
5. The ACE of Soft Skills by Gopalaswamy Ramesh & Mahadevan Ramesh. 2013. Pearson Publishers. New Delhi.
6. Patnaik, P. 2011. *Group Discussion and Interview Skills*. New Delhi: Foundation
7. Sudhir Andrews. 2009. *How to Succeed at Interviews*. New Delhi: Tata McGraw Hill
8. **Sasikumar, V & Dhamija, P.V. 1993. *Spoken English - A Self-Learning Guide to Conversation Practice*. New Delhi: Tata McGraw-Hill**
9. *Dixson, Richard J. Everyday Dialogues in English*. Prentice Hall India Pvt Ltd
10. Mukhopadhyay. L et al. 2012. *Polyskills*. New Delhi: CUP India Pvt Ltd
11. Rizvi, M. A. 2005. *Effective Technical Communication*. New Delhi: Tata McGraw Hill
12. *The Hindu Speaks on Education* by the Hindu Newspaper
13. Naterop, B. Jean and Revell, Rod. 2004. *Telephoning in English*. Cambridge: CUP

**SATAVAHANA UNIVERSITY KARIMNAGAR**  
**MCA II - Year II - SEM**

**PYTHON PROGRAMMING LAB**

**Prerequisites:** Students should install Python on Linux platform.

**Course Objectives:**

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

**Course Outcomes:**

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

**List of Programs:**

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

```

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

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

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

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA II - Year II - SEM****MOBILE APPLICATION DEVELOPMENT LAB**

**Prerequisites --- NIL---**

**Course Objectives**

1. To learn how to develop Applications in android environment.
2. To learn how to develop user interface applications.
3. To learn how to develop URL related applications.

**Course Objectives**

1. Student understands the working of Android OS Practically.
2. Student will be able to develop userinterfaces.
3. Student will be able to develop, deploy and maintain the Android Applications.

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

1. Create an Android application that shows Hello + name of the user and run it on an emulator.  
(b) Create an application that takes the name from a text box and shows hello message along with the name entered in text box, when the user clicks the OK button.
2. Create a screen that has input boxes for User Name, Password, Address, Gender (radio buttons for male and female), Age (numeric), Date of Birth (Date Picket), State (Spinner) and a Submit button. On clicking the submit button, print all the data below the Submit Button. Use (a) Linear Layout (b) Relative Layout and (c) Grid Layout or Table Layout.
3. Develop an application that shows names as a list and on selecting a name it should show the details of the candidate on the next screen with a "Back" button. If the screen is rotated to landscape mode (width greater than height), then the screen should show list on left fragment and details on right fragment instead of second screen with back button. Use Fragment transactions and Rotation event listener.
4. Develop an application that uses a menu with 3 options for dialing a number, opening a website and to send an SMS. On selecting an option, the appropriate action should be invoked using intents.
5. Develop an application that inserts some notifications into Notification area and whenever a notification is inserted, it should show a toast with details of the notification.
6. Create an application that uses a text file to store user names and passwords (tab separated fields and one record per line). When the user submits a login name and password through a screen, the details should be verified with the text file data and if they match, show a dialog saying that login is successful. Otherwise, show the dialog with Login Failed message.
7. Create a user registration application that stores the user details in a database table.
8. Create a database and a user table where the details of login names and passwords are stored. Insert some names and passwords initially. Now the login details entered by the user should be verified with the database and an appropriate dialog should be shown to the user.
9. Create an admin application for the user table, which shows all records as a list and the admin can select any record for edit or modify. The results should be reflected in the table.
10. Develop an application that shows all contacts of the phone along with details like name, phone number, mobile number etc.
11. Create an application that saves user information like name, age, gender etc. in shared preference and retrieves them when the program restarts.

12. Create an alarm that rings every Sunday at 8:00 AM. Modify it to use a time picker to set alarm time.
13. Create an application that shows the given URL (from a text field) in a browser.
14. Develop an application that shows the current location's latitude and longitude continuously as the device is moving (tracking).
15. Create an application that shows the current location on Google maps.

**Note:**

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

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

1. <http://ai2.appinventor.mit.edu>
2. [https://drive.google.com/file/d/0B8rTtW\\_91YclTWF4czdBMEpZcWs/view](https://drive.google.com/file/d/0B8rTtW_91YclTWF4czdBMEpZcWs/view)

**Textbooks:**

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

**References:**

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

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA III - Year I - SEM****DATA ANALYTICS****Course Objectives:**

1. To explore the fundamental concepts of data analytics.
2. To learn the principles and methods of statistical analysis
3. Discover interesting patterns, analyze supervised and unsupervised models and estimate the accuracy of the algorithms.
4. To understand the various search methods and visualization techniques.

**Course Outcomes:** After completion of this course students will be able to

1. Learn basics of R language and learn how to use R to handle the files with data.
2. Understand different files formats like .csv and .txt and learn how access these files.
3. Design Data Architecture
4. Understand various Data Sources

**UNIT - I**

**Data Management:** Design Data Architecture and manage the data for analysis, understand various sources of Data like Sensors/Signals/GPS etc. Data Management, Data Quality(noise, outliers, missing values, duplicate data) and Data Processing & Processing.

**UNIT - II**

**Data Analytics:** Introduction to Analytics, Introduction to Tools and Environment, Application of Modeling in Business, Databases & Types of Data and variables, Data Modeling Techniques, Missing Imputations etc. Need for Business Modeling.

**UNIT - III**

**Regression:** Concepts, Blue property assumptions, Least Square Estimation, Variable Rationalization, and Model Building etc.

Logistic Regression: Model Theory, Model fit Statistics, Model Construction, Analytics applications to various Business Domains etc.

**UNIT - IV**

**Object Segmentation:** Regression Vs Segmentation – Supervised and Unsupervised Learning, Tree Building – Regression, Classification, Overfitting, Pruning and Complexity, Multiple Decision Trees etc. Time Series Methods: Arima, Measures of Forecast Accuracy, STL approach, Extract features from generated model as Height, Average Energy etc and Analyze for prediction

**UNIT - V**

**Data Visualization:** Pixel-Oriented Visualization Techniques, Geometric Projection Visualization Techniques, Icon-Based Visualization Techniques, Hierarchical Visualization Techniques, Visualizing Complex Data and Relations.

**Text Books:**

1. Student's Handbook for Associate Analytics – II, III.
2. Data Mining Concepts and Techniques, Han, Kamber, 3<sup>rd</sup> Edition, Morgan Kaufmann Publishers.

**References:**

1. Introduction to Data Mining, Tan, Steinbach and Kumar, Addison Wisley, 2006.



2. Data Mining Analysis and Concepts, M. Zaki and W. Meira
3. Mining of Massive Datasets, Jure Leskovec Stanford Univ. Anand RajaramanMilliway Labs  
Jeffrey D Ullman Stanford Univ.

**SATAVAHANA UNIVERSITY KARIMNAGAR**  
**MCA III - Year I - SEM**  
**SOFTWARE TESTING METHODOLOGIES**

**Prerequisites**

1. A course on "Software Engineering".

**Course Objectives:**

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

**Course Outcomes:**

1. Design and develop the best test strategies in accordance to the development model.

**UNIT - I**

**Introduction:** Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs

**Flow graphs and Path testing:** Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

**UNIT - II**

**Transaction Flow Testing:** transaction flows, transaction flow testing techniques. Dataflow testing: Basics of 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**

**Paths, Path products and Regular expressions:** path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.

**Logic Based Testing:** overview, decision tables, path expressions, kv charts, specifications.

**UNIT - IV**

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

**UNIT - V**

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

**Text Books:**

1. Software Testing techniques - Baris Beizer, 2<sup>nd</sup> edition, Dreamtech.
2. Software Testing Tools – Dr.K.V.K.K.Prasad, Dreamtech.

**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.

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA III - Year I - SEM****WEB PROGRAMMING****Course Objectives:**

1. To introduce PHP language for server side scripting
2. To introduce XML and processing of XML Data with Java
3. To introduce Server side programming with Java Servlets and JSP

**Course Outcomes:**

1. have understanding of server side scripting with PHP language
2. have understanding of what is XML and how to parse and use **XML** Data with Java
3. To introduce Server side programming with Java Servlets and JSP

**UNIT - I**

**Introduction to PHP:** Declaring variables, data types, arrays, strings, operators, expressions, control structures, functions, Reading data from web form controls like text boxes, radio buttons, lists etc., Handling File Uploads. Connecting to database (MySQL as reference), executing simple queries, handling results, Handling sessions and cookies

**File Handling in PHP:** File operations like opening, closing, reading, writing, appending, deleting etc. on text and binary files, listing directories.

**UNIT - II**

**HTML Common tags-** List, Tables, images, forms, Frames; Cascading Style sheets;

**XML:** Introduction to XML, Defining XML tags, their attributes and values, Document Type Definition, XML Schemes, Document Object Model, XHTML Parsing XML Data – DOM and SAX Parsers in java.

**UNIT - III**

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

**UNIT - IV**

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

**UNIT - V**

**Client side Scripting:** Introduction to Javascript, Javascript language – declaring variables, scope of variables, functions. event handlers (onclick, onsubmit etc.), Document Object Model, Form validation.

**Text Books:**

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

**References:**

1. Web Programming, building internet applications, Chris Bates 2<sup>nd</sup> edition, Wiley Dreamtech
2. Java Server Pages —Hans Bergsten, SPD O'Reilly,
3. Java Script, D.Flanagan
4. Beginning Web Programming-Jon Duckett WROX.

**SATAVAHANA UNIVERSITY KARIMNAGAR**  
**MCA III - Year I - SEM**  
**ETHICAL HACKING (Professional Elective - II)**

**Prerequisites:**

1. A course on "Computer Networks"
2. A course on "Network Security and Cryptography"

**Course Objectives:**

1. To introduce the methodologies and framework of ethical hacking for enhancing the security.

**Course Outcomes:**

1. Gain the knowledge of the use and availability of tools to support an ethical hack
2. Gain the knowledge of interpreting the results of a controlled attack
3. Comprehend the dangers associated with penetration testing

**UNIT - I**

**Introduction:** Hacking Impacts, The Hacker

**Framework:** Planning the test, Sound Operations, Reconnaissance, Enumeration, Vulnerability Analysis, Exploitation, Final Analysis, Deliverable, Integration

**Information Security Models:** Computer Security, Network Security, Service Security, Application Security, Security Architecture

**Information Security Program:** The Process of Information Security, Component Parts of Information Security Program, Risk Analysis and Ethical Hacking

**UNIT - II**

**The Business Perspective:** Business Objectives, Security Policy, Previous Test Results, Business Challenges

**Planning for a Controlled Attack:** Inherent Limitations, Imposed Limitations, Timing is Everything, Attack Type, Source Point, Required Knowledge, Multi-Phased Attacks, Teaming and Attack Structure, Engagement Planner, The Right Security Consultant, The Tester, Logistics, Intermediates, Law Enforcement

**UNIT - III**

**Preparing for a Hack:** Technical Preparation, Managing the Engagement

**Reconnaissance:** Social Engineering, Physical Security, Internet Reconnaissance

**UNIT - IV**

**Enumeration:** Enumeration Techniques, Soft Objective, Looking Around or Attack, Elements of Enumeration, Preparing for the Next Phase

**Exploitation:** Intuitive Testing, Evasion, Threads and Groups, Operating Systems, Password Crackers, RootKits, applications, Wardialing, Network, Services and Areas of Concern

**UNIT - V**

**Deliverable:** The Deliverable, The Document, Overall Structure, Aligning Findings, Presentation

**Integration:** Integrating the Results, Integration Summary, Mitigation, Defense Planning, Incident Management, Security Policy, Conclusion

**Text Book:**

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

**References:**

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

**SATAVAHANA UNIVERSITY KARIMNAGAR**  
**MCA III - Year I - SEM**  
**ADVANCED DATABASES (Professional Elective - II)**

**Prerequisites:**

1. A course on "Database Management Systems"

**Course Objectives:**

1. The purpose of the course is to enrich the previous knowledge of database systems and exposing the need for distributed database technology to confront with the deficiencies of the centralized database systems.
2. Introduce basic principles and implementation techniques of distributed database systems.
3. Equip students with principles and knowledge of parallel and object oriented databases.
4. Topics include distributed DBMS architecture and design; query processing and optimization; distributed transaction management and reliability; parallel and object database management systems.

**Course Outcomes:**

1. Understand theoretical and practical aspects of distributed database systems.
2. Study and identify various issues related to the development of distributed database system.
3. Understand the design aspects of object oriented database system and related development.

**UNIT - I**

**Introduction;** Distributed Data Processing, Distributed Database System, Promises of DDBSs, Problem areas.

**Distributed DBMS Architecture:** Architectural Models for Distributed DBMS, DDMBS Architecture.

**Distributed Database Design:** Alternative Design Strategies, Distribution Design issues, Fragmentation, Allocation.

**UNIT - II**

**Query processing and decomposition:** Query processing objectives, characterization of query processors, layers of query processing, query decomposition, localization of distributed data.

**Distributed query Optimization:** Query optimization, centralized query optimization, distributed query optimization algorithms.

**UNIT - III**

**Transaction Management:** Definition, properties of transaction, types of transactions, distributed concurrency control: serializability, concurrency control mechanisms & algorithms, time - stamped & optimistic concurrency control Algorithms, deadlock Management.

**UNIT –IV**

**Distributed DBMS Reliability:** Reliability concepts and measures, fault-tolerance in distributed systems, failures in Distributed DBMS, local & distributed reliability protocols, site failures and network partitioning.

**Parallel Database Systems:** Parallel database system architectures, parallel data placement, parallel query processing, load balancing, database clusters.

**UNIT - V**

**Distributed object Database Management Systems:** Fundamental object concepts and models, object distributed design, architectural issues, object management, distributed object storage, object query Processing.

**Object Oriented Data Model** : Inheritance, object identity, persistent programming languages, persistence of objects, comparison OODBMS and ORDBMS

**Text Books:**

1. M.Tamer OZSU and Patuck Valduriez: Principles of Distributed Database Systems, Pearson Edn. Asia, 2001.
2. Stefano Ceri and Giuseppe Pelagatti: Distributed Databases, McGraw Hill.

**References:**

1. Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom: "Database Systems: The Complete Book", 2<sup>nd</sup> Edition, Pearson International Edition

**SATAVAHANA UNIVERSITY KARIMNAGAR**  
**MCA III - Year I - SEM**

**INTERNET OF THINGS (Professional Elective - II)**

**Course Objectives:**

1. To introduce the terminology, technology and its applications
2. To introduce the concept of M2M (machine to machine) with necessary protocols
3. To introduce the Python Scripting Language which is used in many IoT devices
4. To introduce the Raspberry PI platform, that is widely used in IoT applications
5. To introduce the implementation of web based services on IoT devices

**Course Outcomes:**

1. Interpret the impact and challenges posed by IoT networks leading to new architectural models.
2. Compare and contrast the deployment of smart objects and the technologies to connect them to network.
3. Appraise the role of IoT protocols for efficient network communication.
4. Elaborate the need for Data Analytics and Security in IoT.
5. Illustrate different sensor technologies for sensing real world entities and identify the applications of IoT in Industry.

**UNIT - I**

**Introduction to Internet of Things:** Definition and Characteristics of IoT, Physical Design of IoT – IoT Protocols, IoT communication models, IoT Communication APIs IoT enabled Technologies – Wireless Sensor Networks, Cloud Computing, Big data analytics, Communication protocols, Embedded Systems, IoT Levels and Templates Domain Specific IoTs – Home, City, Environment, Energy, Retail, Logistics, Agriculture, Industry, health and Lifestyle

**UNIT - II**

**IoT and M2M:** Software defined networks, network function virtualization, difference between SDN and NFV for IoT Basics of IoT System Management with NETCOZF, YANG- NETCONF, YANG, SNMP NETOPEER

**UNIT - III**

**Introduction to Python:** Language features of Python, Data types, data structures, Control of flow, functions, modules, packaging, file handling, data/time operations, classes, Exception handling Python packages - JSON, XML, HTTPLib, URLLib, SMTPLib

**UNIT - IV**

**IoT Physical Devices and Endpoints:** Introduction to Raspberry PI-Interfaces (serial, SPI, I2C) Programming – Python program with Raspberry PI with focus of interfacing external gadgets, controlling output, reading input from pins.

**UNIT - V**

**IoT Physical Servers and Cloud Offerings:** Introduction to Cloud Storage models and communication APIs Webserver – Web server for IoT, Cloud for IoT, Python web application framework Designing a RESTful web API

**Text Books:**

1. Internet of Things - A Hands-on Approach, Arshdeep Bahga and Vijay Madiseti, Universities Press, 2015, ISBN: 9788173719547



2. Getting Started with Raspberry Pi, Matt Richardson & Shawn Wallace, O'Reilly (SPD), 2014, ISBN: 9789350239759

**SATAVAHANA UNIVERSITY KARIMNAGAR**  
**MCA III - Year I - SEM**

**IMAGE PROCESSING (Professional Elective - II)**

**Prerequisites:**

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

**Course Objectives:**

1. Provide a theoretical and mathematical foundation of fundamental Digital Image Processing concepts.
2. The topics include image acquisition; sampling and quantization; preprocessing; enhancement; restoration; segmentation; and compression.

**Course Outcomes:**

1. Demonstrate the knowledge of the basic concepts of two-dimensional signal acquisition, sampling, and quantization.
2. Demonstrate the knowledge of filtering techniques.
3. Demonstrate the knowledge of 2D transformation techniques.
4. Demonstrate the knowledge of image enhancement, segmentation, restoration and compression techniques.

**UNIT - I**

**Digital Image Fundamentals:** Digital Image through Scanner, Digital Camera. Concept of Gray Levels. Gray Level to Binary Image Conversion. Sampling and Quantization. Relationship between Pixels. Imaging Geometry. 2D Transformations-DFT, DCT, KLT and SVD.

**UNIT - II**

Image Enhancement in Spatial Domain Point Processing, Histogram Processing, Spatial Filtering, Enhancement in Frequency Domain, Image Smoothing, Image Sharpening.

**UNIT - III**

Image Restoration Degradation Model, Algebraic Approach to Restoration, Inverse Filtering, Least Mean Square Filters, Constrained Least Squares Restoration, Interactive Restoration.

**UNIT - IV**

Image Segmentation Detection of Discontinuities, Edge Linking and Boundary Detection, Thresholding, Region Oriented Segmentation.

**UNIT - V**

Image Compression Redundancies and their Removal Methods, Fidelity Criteria, Image Compression Models, Source Encoder and Decoder, Error Free Compression, Lossy Compression.

**Text Book:**

1. Digital Image Processing: R.C. Gonzalez & R.E. Woods, 2<sup>nd</sup> Edition, Addison Wesley/ Pearson Education, 2004.

**References:**

1. Fundamentals of Digital Image Processing: A.K.Jain , PHI.
2. Digital Image Processing using MATLAB: Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins: Pearson Education India, 2004.
3. Digital Image Processing: William K. Pratt, John Wiley, 3<sup>rd</sup> Edition, 2004.

**SATAVAHANA UNIVERSITY KARIMNAGAR****MCA III - Year I - SEM****DATA ANALYTICS LAB  
(Data Analytics Using R)****Course Objectives:**

1. To provide an overview of a new language R used for data Analytics.
2. To present the basic techniques for extracting information from large datasets
3. To familiarize students with how various statistics like mean median etc. can be collected for data exploration.
4. Predict outcomes with supervised learning techniques and Unearth the patterns with unsupervised techniques

**Course Outcomes:** After completion of this course students will be able to

1. Learn basics of R language and learn how to use R to handle the files with data.
2. Understand different files formats like .csv and .txt and learn how access these files.
3. Design Data Architecture
4. Understand various Data Sources

**List of Experiments**

1. Demonstrate data cleaning – missing values
2. Implement data normalization (min-max, z-score)
3. Implement attribute subset selection for data reduction
4. Demonstrate outlier detection
5. Perform analytics on any standard data set
6. Implement linear regression
7. Implement logistic regression
8. Construct decision tree for weather data set
9. Analyze time-series data
10. Work on any data visualization tool

**Text Books:**

1. Student's Handbook for Associate Analytics – II, III.
2. Data Mining Concepts and Techniques, Han, Kamber, 3<sup>rd</sup> Edition, Morgan Kaufmann Publishers.

**References:**

1. Introduction to Data Mining, Tan, Steinbach and Kumar, Addison Wesley, 2006.
2. Data Mining Analysis and Concepts, M. Zaki and W. Meira
3. Mining of Massive Datasets, Jure Leskovec Stanford Univ. Anand RajaramanMilliway Labs Jeffrey D Ullman Stanford Univ.

**SATAVAHANA UNIVERSITY KARIMNAGAR**  
**MCA III - Year I - SEM**

**WEB PROGRAMMING & TESTING TOOLS LAB**

**WEB PROGRAMMING LAB:**

**Prerequisites:** A Course on “Objected Oriented Programming through Java”.

**Co-requisites:** A course on “Web Technologies”.

**Course Objectives:**

1. To provide hands-on experience on web technologies
2. To develop client-server application using web technologies
3. To introduce server side programming with Java servlets and JSP

**Course Outcomes:**

1. Design and develop interactive and dynamic web applications using HTML, CSS, JavaScript and XML
2. Apply client-server principles to develop scalable and enterprise web applications.

**List of Experiments:**

1. Write a PHP script to print prime numbers between 1-50.
2. PHP script to
  - a. Find the length of a string.
  - b. Count no of words in a string.
  - c. Reverse a string.
  - d. Search for a specific string.
3. Write a PHP script to merge two arrays and sort them as numbers, in descending order.
4. Write a PHP script that reads data from one file and write into another file.
5. Develop static pages (using Only HTML) of an online book store. The pages should resemble: [www.amazon.com](http://www.amazon.com). The website should consist the following pages.
  - a. Home page
  - b. Registration and user Login
  - c. User Profile Page
  - d. Books catalog
  - e. Shopping Cart
  - f. Payment By credit card
  - g. Order Conformation
6. Validate the Registration, user login, user profile and payment by credit card pages using JavaScript.
7. Create and save an XML document on the server, which contains 10 users information. Write a program, which takes User Id as an input and returns the user details by taking the user information from the XML document.
8. Install TOMCAT web server. Convert the static web pages of assignments 2 into dynamic web pages using servlets and cookies. Hint: Users information (user id, password, credit card number) would be stored in web.xml. Each user should have a separate Shopping Cart.
9. Redo the previous task using JSP by converting the static web pages of assignments 2 into dynamic web pages. Create a database with user information and books information. The books catalogue should be dynamically loaded from the database. Follow the MVC architecture while doing the website.

**Text Books:**

1. Web Technologies: A Computer Science Perspective, Jeffrey C. Jackson, Pearson Education

**References:**

1. Deitel H.M. and Deitel P.J., "Internet and World Wide Web How to program", Pearson International, 2012, 4<sup>th</sup> Edition.
2. J2EE: The complete Reference By James Keogh, McGraw-Hill
3. Bai and Ekedhi, The Web Warrior Guide to Web Programming, Thomson
4. Paul Dietel and Harvey Deitel, "Java How to Program", Prentice Hall of India, 8<sup>th</sup> Edition
5. Web technologies, Black Book, Dreamtech press.
6. Gopalan N.P. and Akilandeswari J., "Web Technology", Prentice Hall of India

**TESTING TOOLS LAB:**

**Prerequisites:** basic knowledge of programming.

**Course Objectives:**

1. To provide knowledge of Software Testing Methods .
2. To develop skills in software test automation and management using latest tools.

**Course Outcomes:**

1. Design and develop the best test strategies in accordance to the development model.

**USING WINRUNNER-CASE TOOL**

1. Recording in context sensitive mode and analog mode
2. GUI checkpoint for single property
3. GUI checkpoint for single object/window
4. GUI checkpoint for multiple objects
5. a) Bitmap checkpoint for object/window  
b) Bitmap checkpoint for screen area
6. Database checkpoint for Default check
7. Database checkpoint for custom check
8. Database checkpoint for runtime record check
9. a) Data driven test for dynamic test data submission  
b) Data driven test through flat files  
c) Data driven test through front grids  
d) Data driven test through excel test
10. a) Batch testing without parameter passing  
b) Batch testing with parameter passing  
c) Data driven batch
11. Silent mode test execution without any interruption
12. Test case for calculator in windows application

**Text Books:**

1. A Practitioner's Guide to Software Test Design, Lee Copeland, 2003
2. The Art of Software Testing, 2nd edition, Glenford Myers, et. el., 2004
3. Software Testing Techniques, 2nd edition, Boris Beizer, 1990
4. How to Break Software: A Practical Guide to Testing, James Whittaker, 2002
5. Testing Object-Oriented Systems: Models, Patterns, and Tools, Robert V. Binder, 1999

**References:**

1. <http://www.onestoptesting.com/win-runner/b.asp>

**SATAVAHANA UNIVERSITY KARIMNAGAR**  
**MCA III - Year I - SEM**  
**PROJECT PHASE - I**

**Prerequisites:** None

**Course Objectives:**

1. To identify a problem, analyse, design and code
2. To demonstrate with sufficient case studies

**Course Outcomes:** At the end of the course the student will be able to:

1. Ability to Synthesize and apply prior knowledge to designing and implementing solutions to open-ended computational problems while considering multiple realistic constraints.

**SATAVAHANA UNIVERSITY KARIMNAGAR**  
**MCA III - Year II - SEM**

**PROJECT PHASE - II**

**Prerequisites:** None

**Course Objectives:**

1. To identify a problem, analyse, design and code
2. To demonstrate with sufficient case studies

**Course Outcomes:** At the end of the course the student will be able to:

1. Ability to Synthesize and apply prior knowledge to designing and implementing solutions to open-ended computational problems while considering multiple realistic constraints.
2. Ability to design and develop the software with SE practices and standards
3. Ability to analyze database, network and application design methods
4. Ability to evaluate the various validation and verification methods
5. Ability to practice CASE tools for solving case studies
6. Ability to analyzing professional issues, including ethical, legal and security issues, related to computing projects.