

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD**  
**MASTER OF COMPUTER APPLICATIONS (MCA)**  
**Effective from Academic Year 2022-23 Admitted Batch**

**COURSE STRUCTURE & SYLLABUS (R22)**

**I YEAR I SEMESTER**

Category	Course Title	L	T	P	Credits
Core Course - I	Mathematical Foundations of Computer Science	3	0	0	3
Core Course - II	C Programming and Data Structures	3	1	0	4
Core Course – III	Python Programming	3	0	0	3
Core Course – IV	Operating Systems	3	0	0	3
Core Course – V	Accounting and Financial Management	3	0	0	3
Laboratory – I	C Programming and Data Structures Lab	0	0	3	1.5
Laboratory – II	Python Programming Lab	0	0	2	1
Laboratory – III	Operating Systems Lab	0	0	3	1.5
	<b>Total Credits</b>	<b>15</b>	<b>1</b>	<b>8</b>	<b>20</b>

**I YEAR II SEMESTER**

Category	Course Title	L	T	P	Credits
Core Course - I	Java Programming	3	0	0	3
Core Course - II	Database Management Systems	3	0	0	3
Core Course – III	Computer Networks	3	0	0	3
Core Course – IV	Data Mining	3	0	0	3
Core Course – V	Software Engineering	3	0	0	3
Laboratory – I	Database Management Systems Lab	0	0	4	2
Laboratory – II	Computer Networks Lab	0	0	3	1.5
Laboratory – III	Java Programming Lab	0	0	3	1.5
	<b>Total Credits</b>	<b>15</b>	<b>0</b>	<b>10</b>	<b>20</b>

**II YEAR I SEMESTER**

Category	Course Title	L	T	P	Credits
Core Course - I	Enterprise Cloud Concepts	3	0	0	3
Core Course - II	Full Stack Development	3	0	0	3
Core Course – III	Machine Learning	3	0	0	3
Professional Elective - I	<b>Professional Elective - I</b> Artificial Intelligence Information Retrieval Systems Ad-hoc and Sensor Networks	3	0	0	3
Professional Elective - II	<b>Professional Elective - II</b> Cyber Security Mobile Computing Software Testing Methodologies	3	0	0	3
Laboratory – I	Enterprise Cloud Concepts Lab	0	0	3	1.5
Laboratory – II	Full Stack Development Lab	0	0	3	1.5
Laboratory – III	Machine Learning Lab	0	1	2	2
	<b>Total Credits</b>	<b>15</b>	<b>1</b>	<b>8</b>	<b>20</b>

**II YEAR II SEMESTER**

<b>Category</b>	<b>Course Title</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>Credits</b>
Professional Elective - III	<b>Professional Elective - III</b> Internet of Things Mobile Application Development Big Data Analytics	3	0	0	3
Open Elective - I	<b>Open Elective - I</b> e-Commerce Cyber Laws & Ethics Management Information System Entrepreneurship	3	0	0	3
Seminar	Seminar	0	0	4	2
Project	Project Work Review - III	0	0	4	2
Project	Project Viva-Voce	0	0	20	10
	<b>Total Credits</b>	<b>6</b>	<b>0</b>	<b>24</b>	<b>20</b>

**\*Note:** Please refer to Academic Regulations for Project Work Review – I & II.

**MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE****MCA I Year I Sem.**

L	T	P	C
3	0	0	3

**Pre-requisites:** An understanding of Math in general is sufficient.**Course Objectives:** To learn

- Introduces the elementary discrete mathematics for computer science and engineering.
- 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:** After learning the contents of this paper the student must be able to

- Ability to understand and construct precise mathematical proofs.
- Ability to use logic and set theory to formulate precise statements.
- Ability to analyze and solve counting problems on finite and discrete structures.
- Ability to describe and manipulate sequences.
- 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.

**TEXT BOOKS:**

1. Discrete Mathematical Structures with Applications to Computer Science: J.P. Tremblay, R. Manohar, McGraw-Hill, 1<sup>st</sup> ed.
2. Discrete Mathematics for Computer Scientists & Mathematicians: Joe I. Mott, Abraham Kandel, Teodore P. Baker, Prentis Hall of India, 2<sup>nd</sup> ed.

**REFERENCES:**

1. Discrete and Combinatorial Mathematics - an applied introduction: Ralph.P. Grimald, Pearson education, 5<sup>th</sup> edition.
2. Discrete Mathematical Structures: Thomas Kosy, Tata McGraw Hill publishing co.

**C PROGRAMMING AND DATA STRUCTURES****MCA I Year I Sem.**

L	T	P	C
3	1	0	4

**Pre-requisites:**

1. Requires analytical skills and logical reasoning.

**Course Objectives:** To learn

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

**Course Outcomes:** After learning the contents of this paper the student must be able to

- 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.
- 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, Pre-processor 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.

**TEXT BOOKS:**

1. C Programming & Data Structures, B.A.Forouzan and R.F. Gilberg,3rd Edition, Cengage Learning.
2. Problem Solving and Program Design in C, J.R. Hanly and E.B. Koffman, 5th 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 Theraja, Oxford University Press.

**PYTHON PROGRAMMING****MCA I Year I Sem.**

L	T	P	C
3	0	0	3

**Prerequisites:**

1. Requires analytical skills and logical reasoning.

**Course Objectives:**

1. To learn syntax and semantics and create functions in Python.
2. To handle strings and files in Python.
3. To understand lists, dictionaries and regular expressions in Python.
4. To implement object-oriented programming concepts in Python.
5. To build web services and introduce network and database programming in Python.

**Course Outcomes:**

1. Demonstrate proficiency in handling Strings and File Systems.
2. Create, run and manipulate Python programs using core data structures like Lists, Dictionaries and use Regular Expressions.
3. Interpret the concepts of GUI Programming as used in Python.
4. Implement 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, 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, Exceptions and the sys Module, Related 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, WebSurfing 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

**TEXT BOOK:**

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

**REFERENCES:**

1. Python for Programmers, Paul Deitel, Harvey Deitel, Pearson.
2. Think Python, Allen Downey, Green Tea Press
3. Introduction to Python, Kenneth A. Lambert, Cengage
4. Python Programming: A Modern Approach, Vamsi Kurama, Pearson
5. Learning Python, Mark Lutz, O'Reilly

**OPERATING SYSTEMS****MCA I Year I Sem.**

L	T	P	C
3	0	0	3

**Prerequisites:**

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

**Course Objectives:**

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

**Course Outcomes:**

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

**UNIT - I**

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

**Process** - Process concepts and scheduling, Operations on processes, Cooperating Processes, Threads

**UNIT - II**

**CPU Scheduling** - Scheduling Criteria, Scheduling Algorithms, Multiple -Processor Scheduling. System call interface for process management-fork, exit, wait, waitpid, exec

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

**UNIT - III**

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

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

**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.

**TEXT BOOKS:**

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

**REFERENCES:**

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

**ACCOUNTING AND FINANCIAL MANAGEMENT****MCA I Year I Sem.**

L	T	P	C
3	0	0	3

**Course Objectives:** To learn

- To learn Financial Management and Accounting
- To learn different types of costing

**Course Outcomes:** After learning the contents of this paper the student must be able to

- Able to prepare balance sheets of budget.
- 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.**TEXT BOOKS:**

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



**COMPUTER PROGRAMMING & DATA STRUCTURES LAB****MCA I Year I Sem.**

L	T	P	C
0	0	3	1.5

**Prerequisites:**

1. Requires analytical skills and logical reasoning

**Course Objectives:**

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

**Course Outcomes:**

- 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.
- 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
------------------	-------------------

**TEXT BOOKS:**

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

**PYTHON PROGRAMMING LAB****MCA I Year I Sem.**

L	T	P	C
0	0	2	1

**Course Objectives:**

- To install and run the Python interpreter
- To learn control structures.
- To Understand Lists, Dictionaries in python
- To Handle Strings and Files in Python

**Course Outcomes:** After completion of the course, the student should be able to

- Develop the application specific codes using python.
- Understand Strings, Lists, Tuples and Dictionaries in Python
- Verify programs using modular approach, file I/O, Python standard library
- Implement Digital Systems using Python

Note: The lab experiments will be like the following experiment examples

**Week -1:**

1. i) Use a web browser to go to the Python website <http://python.org>. This page contains information about Python and links to Python-related pages, and it gives you the ability to search the Python documentation.  
ii) Start the Python interpreter and type help() to start the online help utility.
2. Start a Python interpreter and use it as a Calculator.
3.
  - i) Write a program to calculate compound interest when principal, rate and number of periods are given.
  - ii) Given coordinates (x1, y1), (x2, y2) find the distance between two points
4. Read name, address, email and phone number of a person through keyboard and print the details.

**Week - 2:**

1. Print the below triangle using for loop.  

```

5
4 4
3 3 3
2 2 2 2
1 1 1 1 1

```
2. Write a program to check whether the given input is digit or lowercase character or uppercase character or a special character (use 'if-else-if' ladder)
3. Python Program to Print the Fibonacci sequence using while loop
4. Python program to print all prime numbers in a given interval (use break)

**Week - 3:**

1. i) Write a program to convert a list and tuple into arrays.  
ii) Write a program to find common values between two arrays.
2. Write a function called gcd that takes parameters a and b and returns their greatest common divisor.
3. Write a function called palindrome that takes a string argument and returns True if it is a palindrome and False otherwise. Remember that you can use the built-in function len to check the length of a string.

**Week - 4:**

1. Write a function called is\_sorted that takes a list as a parameter and returns True if the list is sorted in ascending order and False otherwise.
2. Write a function called has\_duplicates that takes a list and returns True if there is any element that appears more than once. It should not modify the original list.
  - i). Write a function called remove\_duplicates that takes a list and returns a new list with only the unique elements from the original. Hint: they don't have to be in the same order.
  - ii). The wordlist I provided, words.txt, doesn't contain single letter words. So you might want to add "l", "a", and the empty string.
  - iii). Write a python code to read dictionary values from the user. Construct a function to invert its content. i.e., keys should be values and values should be keys.
3.
  - i) Add a comma between the characters. If the given word is 'Apple', it should become 'A,p,p,l,e'
  - ii) Remove the given word in all the places in a string?

- iii) Write a function that takes a sentence as an input parameter and replaces the first letter of every word with the corresponding upper case letter and the rest of the letters in the word by corresponding letters in lower case without using a built-in function?
4. Writes a recursive function that generates all binary strings of n-bit length

**Week - 5:**

1. i) Write a python program that defines a matrix and prints  
ii) Write a python program to perform addition of two square matrices  
iii) Write a python program to perform multiplication of two square matrices
2. How do you make a module? Give an example of construction of a module using different geometrical shapes and operations on them as its functions.
3. Use the structure of exception handling all general purpose exceptions.

**Week-6:**

1. a. Write a function called draw\_rectangle that takes a Canvas and a Rectangle as arguments and draws a representation of the Rectangle on the Canvas.  
b. Add an attribute named color to your Rectangle objects and modify draw\_rectangle so that it uses the color attribute as the fill color.  
c. Write a function called draw\_point that takes a Canvas and a Point as arguments and draws a representation of the Point on the Canvas.  
d. Define a new class called Circle with appropriate attributes and instantiate a few Circle objects. Write a function called draw\_circle that draws circles on the canvas.
2. Write a Python program to demonstrate the usage of Method Resolution Order (MRO) in multiple levels of Inheritances.
3. Write a python code to read a phone number and email-id from the user and validate it for correctness.

**Week- 7**

1. Write a Python code to merge two given file contents into a third file.
2. Write a Python code to open a given file and construct a function to check for given words present in it and display on found.
3. Write a Python code to Read text from a text file, find the word with most number of occurrences
4. Write a function that reads a file *file1* and displays the number of words, number of vowels, blank spaces, lower case letters and uppercase letters.

**Week - 8:**

1. Import numpy, Plotpy and Scipy and explore their functionalities.
2. a) Install NumPy package with pip and explore it.
3. Write a program to implement Digital Logic Gates – AND, OR, NOT, EX-OR
4. Write a program to implement Half Adder, Full Adder, and Parallel Adder
5. Write a GUI program to create a window wizard having two text labels, two text fields and two buttons as Submit and Reset.

**TEXT BOOKS:**

1. Supercharged Python: Take your code to the next level, Overland
2. Learning Python, Mark Lutz, O'reilly

**REFERENCES:**

1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson
2. Python Programming A Modular Approach with Graphics, Database, Mobile, and Web Applications, Sheetal Taneja, Naveen Kumar, Pearson
3. Programming with Python, A User's Book, Michael Dawson, Cengage Learning, India Edition
4. Think Python, Allen Downey, Green Tea Press
5. Core Python Programming, W. Chun, Pearson
6. Introduction to Python, Kenneth A. Lambert, Cengage

**OPERATING SYSTEMS LAB****MCA I Year I Sem.**

L	T	P	C
0	0	3	1.5

**Prerequisites:**

- A course on “Programming for Problem Solving”.
- A course on “Computer Organization and Architecture”.

**Co-requisite:**

- A course on “Operating Systems”.

**Course Objectives:**

- To provide an understanding of the design aspects of operating system concepts through simulation
- Introduce basic Unix commands, system call interface for process management, interprocess communication and I/O in Unix

**Course Outcomes:**

- Simulate and implement operating system concepts such as scheduling, deadlock management, file management and memory management.
- Able to implement C programs using Unix system calls

**List of Experiments:**

1. Write C programs to simulate the following CPU Scheduling algorithms  
a) FCFS b) SJF c) Round Robin d) priority
2. Write programs using the I/O system calls of UNIX/LINUX operating system  
(open, read, write, close, fcntl, seek, stat, opendir, readdir)
3. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance and Prevention.
4. Write a C program to implement the Producer – Consumer problem using semaphores using UNIX/ LINUX system calls.
5. Write C programs to illustrate the following IPC mechanisms  
a) Pipes b) FIFOs c) Message Queues d) Shared Memory
6. Write C programs to simulate the following memory management techniques  
a) Paging b) Segmentation
7. Write C programs to simulate Page replacement policies  
a) FCFS                      b) LRU c) Optimal

**TEXT BOOKS:**

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, William Stallings, Fifth Edition–2005, Pearson Education/PHI
2. Operating System - A Design Approach-Crowley, TMH.
3. Modern Operating Systems, Andrew S Tanenbaum, 2nd edition, Pearson/PHI
4. UNIX Programming Environment, Kernighan and Pike, PHI/Pearson Education
5. UNIX Internals: The New Frontiers, U. Vahalia, Pearson Education

**JAVA PROGRAMMING**

MCA I Year II Sem.

L	T	P	C
3	0	0	3

**Course Objectives:**

- To introduce object-oriented programming principles and apply them in solving problems.
- To introduce the implementation of packages and interfaces.
- To introduce the concepts of exception handling and multithreading.
- To introduce the design of Graphical User Interface using swing controls.

**Course Outcomes:**

- Able to solve real world problems using OOP techniques.
- Able to solve problems using java collection framework and I/O classes.
- Able to develop multithreaded applications with synchronization.
- Able to design GUI based applications.

**UNIT - I**

**Foundations of Java:** History of Java, Java Features, Variables, Data Types, Operators, Expressions, Control Statements. Elements of Java - Class, Object, Methods, Constructors and Access Modifiers, Generics, Inner classes, String class and Annotations.

**OOP Principles:** Encapsulation – concept, setter and getter method usage, this keyword. Inheritance - concept, Inheritance Types, super keyword. Polymorphism – concept, Method Overriding usage and Type Casting. Abstraction – concept, abstract keyword and Interface.

**UNIT – II**

**Exception Handling:** Exception and Error, Exception Types, Exception Handler, Exception Handling Clauses – try, catch, finally, throws and the throw statement, Built-in-Exceptions and Custom Exceptions.

**Files and I/O Streams:** The file class, Streams, The Byte Streams, Filtered Byte Streams, The Random Access File class.

**UNIT – III**

**Packages-** Defining a Package, CLASSPATH, Access Specifiers, importing packages. Few Utility Classes - String Tokenizer, BitSet, Date, Calendar, Random, Formatter, Scanner.

**Collections:** Collections overview, Collection Interfaces, Collections Implementation Classes, Sorting in Collections, Comparable and Comparator Interfaces.

**UNIT – IV**

**Multithreading:** Process and Thread, Differences between thread-based multitasking and process-based multitasking, Java thread life cycle, creating threads, thread priorities, synchronizing threads, inter thread communication.

**Java Database Connectivity:** Types of Drivers, JDBC architecture, JDBC Classes and Interfaces, Basic steps in Developing JDBC Application, Creating a New Database and Table with JDBC.

**UNIT - V**

**GUI Programming with Swing –** Introduction, limitations of AWT, MVC architecture, components, containers, Layout Manager Classes, Simple Applications using AWT and Swing.

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

**TEXT BOOKS:**

1. Java The complete reference, 9th edition, Herbert Schildt, McGraw Hill Education (India) Pvt. Ltd.
2. Understanding Object-Oriented Programming with Java, updated edition, T. Budd, Pearson Education.

**REFERENCES:**

1. An Introduction to programming and OO design using Java, J. Nino and F.A. Hosch, John Wiley & sons
2. Introduction to Java programming, Y. Daniel Liang, Pearson Education.
3. Object Oriented Programming through Java, P. Radha Krishna, University Press.
4. Programming in Java, S. Malhotra, S. Chudhary, 2nd edition, Oxford Univ. Press.
5. Java Programming and Object-oriented Application Development, R. A. Johnson, Cengage Learning.

**DATABASE MANAGEMENT SYSTEMS****MCA I Year II Sem.**

L	T	P	C
3	0	0	3

**Prerequisites:** A course on “Data Structures”.**Course Objectives:**

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

**Course Outcomes:**

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

**UNIT - I**

**Database System Applications:** 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 database 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 databases.

**Schema Refinement:** Problems caused by redundancy, decompositions, problems related to decomposition, reasoning about functional dependencies, FIRST, SECOND, THIRD normal forms, BCNF, lossless join decomposition, multivalued 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 based Indexing, Comparison of File Organizations, Indexes-Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM), B+ Trees: A Dynamic Index Structure.

**TEXT BOOKS:**

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

**REFERENCES:**

1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7<sup>th</sup> 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.

**COMPUTER NETWORKS****MCA I Year II Sem.**

L	T	P	C
3	0	0	3

**Prerequisites**

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

**Course Objectives**

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

**Course Outcomes**

- Gain the knowledge of the basic computer network technology.
- Gain the knowledge of the functions of each layer in the OSI and TCP/IP reference model.
- Obtain the skills of subnetting and routing mechanisms.
- Familiarity with the essential protocols of computer networks, and how they can be 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.  
 Data link layer: Design issues, framing, Error detection and correction.

**UNIT - II**

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

**UNIT - III**

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

**UNIT - IV**

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

**UNIT - V**

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

**TEXT BOOK:**

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

**REFERENCES:**

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



**DATA MINING****MCA I Year II Sem.**

L	T	P	C
3	0	0	3

**Prerequisites:**

- A course on “Database Management Systems”.

**Course Objectives:**

- Learn data mining concepts understand association rules mining.
- Discuss classification algorithms learn how data is grouped using clustering techniques.
- To develop the abilities of critical analysis to data mining systems and applications.
- To implement practical and theoretical understanding of the technologies for data mining.
- To understand the strengths and limitations of various data mining models.

**Course Outcomes:**

- Ability to perform the preprocessing of data and apply mining techniques on it.
- Ability to identify the association rules, classification and clusters in large data sets.
- Ability to solve real world problems in business and scientific information using data mining.
- Ability to classify web pages, extracting knowledge from the web.

**UNIT-I:**

**Introduction to Data Mining:** Introduction, What is Data Mining, Definition, KDD, Challenges, Data Mining Tasks, Data Pre-processing, Data Cleaning, Missing data, Dimensionality Reduction, Feature Subset Selection, Discretization and Binarization, Data Transformation; Measures of Similarity and Dissimilarity- Basics.

**UNIT-II:**

**Association Rules:** Problem Definition, Frequent Item Set Generation, The APRIORI Principle, Support and Confidence Measures, Association Rule Generation; APRIORI Algorithm, The Partition Algorithms, FP-Growth Algorithms, Compact Representation of Frequent Item Set- Maximal Frequent Item Set, Closed Frequent Item Set.

**UNIT-III:**

**Classification:** Problem Definition, General Approaches to solving a classification problem, Evaluation of Classifiers, Classification techniques, Decision Trees-Decision tree Construction, Methods for Expressing attribute test conditions, Measures for Selecting the Best Split, Algorithm for Decision tree Induction; Naive-Bayes Classifier, Bayesian Belief Networks; K- Nearest neighbor classification-Algorithm and Characteristics.

**UNIT-IV:**

**Clustering:** Problem Definition, Clustering Overview, Evaluation of Clustering Algorithms, Partitioning Clustering-K-Means Algorithm, K-Means Additional issues, PAM Algorithm; Hierarchical Clustering-Agglomerative Methods and divisive methods, Basic Agglomerative Hierarchical Clustering Algorithm, Specific techniques, Key Issues in Hierarchical Clustering, Strengths and Weakness; Outlier Detection.

**UNIT-V:**

**Web and Text Mining:** Introduction, web mining, web content mining, web structure mining, we usage mining, Text mining –unstructured text, episode rule discovery for texts, hierarchy of categories, text clustering.

**TEXTBOOKS:**

1. Data Mining- Concepts and Techniques- Jiawei Han, Micheline Kamber, Morgan Kaufmann Publishers, Elsevier, Edition, 2006.
2. Introduction to Data Mining, Pang-Ning Tan, Vipin Kumar, Michael Steinbach, Pearson Education.
3. Data mining Techniques and Applications, Hongbo Du Cengage India Publishing.

**REFERENCES:**

1. Data Mining Techniques, Arun K Pujari, 3rd Edition, Universities Press.
2. Data Mining Principles & Applications – T.V Sveresh Kumar, B.Esware Reddy, Jagadish S Kalimani, Elsevier.
3. Data Mining, Vikaram Pudi, P Radha Krishna, Oxford University Press.

**SOFTWARE ENGINEERING****MCA I Year II Sem.**

L	T	P	C
3	0	0	3

**Course Objectives**

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

**Course Outcomes**

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

**UNIT - I**

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

**A Generic view of process:** Software engineering- a layered technology, a process framework, the capability maturity model integration (CMMI)

**Process models:** The waterfall model, Spiral model and Agile methodology

**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.

**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.

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

**UNIT - V**

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

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

**TEXT BOOKS:**

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, McGraw Hill International Edition.
2. Software Engineering- Sommerville, 7<sup>th</sup> edition, Pearson Education.

**REFERENCES:**

1. The unified modeling language user guide Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education.
2. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley.
3. Software Engineering principles and practice- Waman S Jawadkar, The McGraw-Hill Companies.
4. Fundamentals of object-oriented design using UML Meiler page-Jones: Pearson Education.

**DATABASE MANAGEMENT SYSTEMS LAB****MCA I Year II Sem.**

L	T	P	C
0	0	4	2

**Co-requisites:**

- Co-requisite of course “Database Management Systems”

**Course Objectives:**

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

**Course Outcomes:**

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

**List of Experiments:**

1. Concept design with E-R Model
2. Relational Model
3. Normalization
4. Practicing DDL commands
5. Practicing DML commands
6. A. Querying (using ANY, ALL, UNION, INTERSECT, JOIN, Constraints etc.)  
B. Nested, Correlated subqueries
7. Queries using Aggregate functions, GROUP BY, HAVING and Creation and dropping of Views.
8. Triggers (Creation of insert trigger, delete trigger, update trigger)
9. Procedures
10. Usage of Cursors

**TEXT BOOKS:**

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

**REFERENCES:**

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

**COMPUTER NETWORKS LAB**

MCA I Year II Sem.

L	T	P	C
0	0	3	1.5

**Course Objectives**

- To understand the working principle of various communication protocols.
- To understand the network simulator environment and visualize a network topology and observe its performance
- To analyze the traffic flow and the contents of protocol frames

**Course Outcomes**

- Implement data link layer framing methods
- Analyze error detection and error correction codes.
- Implement and analyze routing and congestion issues in network design.
- Implement Encoding and Decoding techniques used in presentation layer
- To be able to work with different network tools

**List of Experiments**

1. Implement the data link layer framing methods such as character, character-stuffing and bit stuffing.
2. Write a program to compute CRC code for the polynomials CRC-12, CRC-16 and CRC CCIP
3. Develop a simple data link layer that performs the flow control using the sliding window protocol, and loss recovery using the Go-Back-N mechanism.
4. Implement Dijkstra's algorithm to compute the shortest path through a network
5. Take an example subnet of hosts and obtain a broadcast tree for the subnet.
6. Implement distance vector routing algorithm for obtaining routing tables at each node.
7. Implement data encryption and data decryption
8. Write a program for congestion control using Leaky bucket algorithm.
9. Write a program for frame sorting techniques used in buffers.
10. **Wireshark**
  - i. Packet Capture Using Wire shark
  - ii. Starting Wire shark
  - iii. Viewing Captured Traffic
  - iv. Analysis and Statistics & Filters.
11. How to run Nmap scan
12. Operating System Detection using Nmap
13. Do the following using NS2 Simulator
  - i. NS2 Simulator-Introduction
  - ii. Simulate to Find the Number of Packets Dropped
  - iii. Simulate to Find the Number of Packets Dropped by TCP/UDP
  - iv. Simulate to Find the Number of Packets Dropped due to Congestion
  - v. Simulate to Compare Data Rate & Throughput.
  - vi. Simulate to Plot Congestion for Different Source/Destination
  - vii. Simulate to Determine the Performance with respect to Transmission of Packets

**TEXT BOOK:**

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

**REFERENCES:**

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

**JAVA PROGRAMMING LAB****MCA I Year II Sem.**

L	T	P	C
0	0	3	1.5

**Course Objectives:**

- To understand OOP principles.
- To understand the Exception Handling mechanism.
- To understand Java collection framework.
- To understand multithreaded programming.
- To understand swing controls in Java.

**Course Outcomes:**

- Able to write the programs for solving real world problems using Java OOP principles.
- Able to write programs using Exceptional Handling approach.
- Able to write multithreaded applications.
- Able to write GUI programs using swing controls in Java.

**List of Experiments:**

1. Use Eclipse or Net bean platform and acquaint yourself 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.
2. Write a Java program to demonstrate the OOP principles. [i.e., Encapsulation, Inheritance, Polymorphism and Abstraction]
3. Write a Java program to handle checked and unchecked exceptions. Also, demonstrate the usage of custom exceptions in real time scenario.
4. Write a Java program on Random Access File class to perform different read and write operations.
5. Write a Java program to demonstrate the working of different collection classes. [Use package structure to store multiple classes].
6. Write a program to synchronize the threads acting on the same object. [Consider the example of any reservations like railway, bus, movie ticket booking, etc.]
7. Write a program to perform CRUD operations on the student table in a database using JDBC.
8. 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 divided by zero.
9. 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. [Use Adapter classes]

**REFERENCES:**

1. Java for Programmers, P. J. Deitel and H. M. Deitel, 10th Edition Pearson education.
2. Thinking in Java, Bruce Eckel, Pearson Education.
3. Java Programming, D. S. Malik and P. S. Nair, Cengage Learning.
4. Core Java, Volume 1, 9th edition, Cay S. Horstmann and G Cornell, Pearson.

**ENTERPRISE CLOUD CONCEPTS****MCA II Year I Sem.**

L	T	P	C
3	0	0	3

**Course Objectives:** Knowledge on significance of cloud computing and its fundamental concepts and models.

**Course Outcomes:**

- Understand importance of cloud architecture
- Illustrating the fundamental concepts of cloud security
- Analyze various cloud computing mechanisms
- Understanding the architecture and working of cloud computing.

**UNIT-I**

**Understanding Cloud Computing:** Origins and influences, Basic Concepts and Terminology, Goals and Benefits, Risks and Challenges.

**Fundamental Concepts and Models:** Roles and Boundaries, Cloud Characteristics, Cloud Delivery Models, Cloud Deployment Models.

**UNIT-II**

**Cloud-Enabling Technology:** Broadband Networks and Internet Architecture, Data Center Technology, Virtualization Technology

**Cloud Computing Mechanisms, Cloud Infrastructure Mechanisms:** Logical Network Perimeter, Virtual Server, Cloud Storage Device, Cloud Usage Monitor, Resource Replication

**UNIT-III**

**Cloud Management Mechanisms:** Remote Administration System, Resource Management System, SLA Management System, Billing Management System, Case Study Example

**Cloud Computing Architecture**

**Fundamental Cloud Architectures:** Workload Distribution Architecture, Resource Pooling Architecture, Dynamic Scalability Architecture, Elastic Resource Capacity Architecture, Service Load Balancing Architecture, Cloud Bursting Architecture, Elastic Disk Provisioning Architecture, Redundant Storage Architecture, Case Study Example

**UNIT-IV**

**Cloud-Enabled Smart Enterprises:** Introduction, Revisiting the Enterprise Journey, Service-Oriented Enterprises, Cloud Enterprises, Smart Enterprises, The Enabling Mechanisms of Smart Enterprises

**Cloud-Inspired Enterprise Transformations:** Introduction, The Cloud Scheme for Enterprise Success, Elucidating the Evolving Cloud Idea, Implications of the Cloud on Enterprise Strategy, Establishing a Cloud-Incorporated Business Strategy

**UNIT-V**

**Transitioning to Cloud-Centric Enterprises:** The Tuning Methodology, Contract Management in the Cloud

**Cloud-Instigated IT Transformations:** Introduction, Explaining Cloud Infrastructures, A Briefing on Next-Generation Services, Service Infrastructures, Cloud Infrastructures, Cloud Infrastructure Solutions, Clouds for Business Continuity, The Relevance of Private Clouds, The Emergence of Enterprise Clouds

**TEXT BOOKS:**

1. Erl Thomas, Puttini Ricardo, Mahmood Zaigham, Cloud Computing: Concepts, Technology & Architecture 1st Edition,
2. Pethuru Raj, Cloud Enterprise Architecture, CRC Press

**REFERENCE:**

1. James Bond, The Enterprise Cloud, O'Reilly Media, Inc.

**FULL STACK DEVELOPMENT****MCA II Year I Sem.**

L	T	P	C
3	0	0	3

**Course Objectives:**

- To learn the core concepts of both the frontend and backend programming course. 2. To get familiar with the latest web development technologies.
- To learn all about databases.
- To learn complete web development process
- To provide an in-depth study of the various web development tools

**Course Outcomes:**

- Develop a fully functioning website and deploy on a web server.
- Gain Knowledge about the front end and back end Tools
- Find and use code packages based on their documentation to produce working results in a project.
- Create web pages that function using external data.

**UNIT-I:**

Web Development Basics: Web development Basics - HTML & Web servers Shell - UNIX CLI Version control - Git & Github HTML, CSS

**UNIT-II:**

Frontend Development: Javascript basics OOPS Aspects of JavaScript Memory usage and Functions in JS AJAX for data exchange with server jQuery Framework jQuery events, UI components etc. JSON data format.

**UNIT-III:**

REACT JS: Introduction to React React Router and Single Page Applications React Forms, Flow Architecture and Introduction to Redux More Redux and Client-Server Communication

**UNIT-IV:**

Java Web Development: JAVA PROGRAMMING BASICS, Model View Controller (MVC) Pattern MVC Architecture using Spring RESTful API using Spring Framework Building an application using Maven

**UNIT-V:**

Databases & Deployment: Relational schemas and normalization Structured Query Language (SQL) Data persistence using Spring JDBC Agile development principles and deploying application in Cloud

**TEXT BOOKS:**

1. Web Design with HTML, CSS, JavaScript and JQuery Set Book by Jon Duckett Professional JavaScript for Web Developers Book by Nicholas C. Zakas
2. Learning PHP, MySQL, JavaScript, CSS & HTML5: A Step-by-Step Guide to Creating Dynamic Websites by Robin Nixon
3. Full Stack JavaScript: Learn Backbone.js, Node.js and MongoDB. Copyright © 2015 BY Azat Mardan

**REFERENCES:**

1. Full-Stack JavaScript Development by Eric Bush.
2. Mastering Full Stack React Web Development Paperback – April 28, 2017 by Tomasz Dyl, Kamil Przeorski, Maciej Czarnecki

**MACHINE LEARNING****MCA II Year I Sem.**

L	T	P	C
3	0	0	3

**Course OBJECTIVES:**

- To introduce students to the basic concepts and techniques of Machine Learning.
- To have a thorough understanding of the Supervised and Unsupervised learning techniques
- To study the various probability based learning techniques
- To understand graphical models of machine learning algorithms

**Course Outcomes:** Upon completion of the course, the students will be able to:

- Distinguish between, supervised, unsupervised and semi-supervised learning
- Apply the apt machine learning strategy for any given problem
- Suggest supervised, unsupervised or semi-supervised learning algorithms for any given problem
- Design systems that use the appropriate graph models of machine learning
- Modify existing machine learning algorithms to improve classification efficiency

**UNIT I: Introduction:**

Learning – Types of Machine Learning – Supervised Learning – The Brain and the Neuron – Design a Learning System – Perspectives and Issues in Machine Learning – Concept Learning Task – Concept Learning as Search – Finding a Maximally Specific Hypothesis – Version Spaces and the Candidate Elimination Algorithm – Linear Discriminants: – Perceptron – Linear Separability – Linear Regression.

**UNIT II: Linear Models:**

Multi-layer Perceptron– Going Forwards – Going Backwards: Back Propagation Error – Multi-layer Perceptron in Practice – Examples of using the MLP – Overview – Deriving Back-Propagation – Radial Basis Functions and Splines – Concepts – RBF Network – Curse of Dimensionality – Interpolations and Basis Functions – Support Vector Machines

**UNIT III: Tree and Probabilistic Models:**

Learning with Trees – Decision Trees – Constructing Decision Trees – Classification and Regression Trees – Ensemble Learning – Boosting – Bagging – Different ways to Combine Classifiers – Basic Statistics – Gaussian Mixture Models – Nearest Neighbor Methods – Unsupervised Learning – K means Algorithms

**UNIT IV: Dimensionality Reduction and Evolutionary Models:**

Dimensionality Reduction – Linear Discriminant Analysis – Principal Component Analysis – Factor Analysis – Independent Component Analysis – Locally Linear Embedding – Isomap – Least Squares Optimization – Evolutionary Learning – Genetic algorithms – Genetic Offspring: - Genetic Operators – Using Genetic Algorithms – Reinforcement Learning – Overview – Getting Lost Example

**UNIT V: Graphical Models:**

Markov Chain Monte Carlo Methods – Sampling – Proposal Distribution – Markov Chain Monte Carlo – Graphical Models – Bayesian Networks – Markov Random Fields – Hidden Markov Models – Tracking Methods

**TEXT BOOKS:**

1. Stephen Marsland, —Machine Learning – An Algorithmic Perspective, Second Edition, Chapman and Hall/ CRC Machine Learning and Pattern Recognition Series, 2014.
2. Tom M Mitchell, —Machine Learning, First Edition, McGraw Hill Education, 2013.

**REFERENCES:**

1. Peter Flach, —Machine Learning: The Art and Science of Algorithms that Make Sense of Data, First Edition, Cambridge University Press, 2012.
2. Jason Bell, —Machine learning – Hands on for Developers and Technical Professionals, First Edition, Wiley, 2014
3. Ethem Alpaydin, —Introduction to Machine Learning 3e (Adaptive Computation and Machine Learning Series), Third Edition, MIT Press, 2014



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

MCA II Year I Sem.

L	T	P	C
3	0	0	3

**Prerequisites:**

- A course on “Computer Programming and Data Structures”.
- A course on “Advanced Data Structures”.
- A course on “Design and Analysis of Algorithms”.
- A course on “Mathematical Foundations of Computer Science”.
- Some background in linear algebra, data structures and algorithms, and probability will all be helpful.

**Course Objectives:**

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

**Course Outcomes:**

- Ability to formulate an efficient problem space for a problem expressed in natural language.
- Select a search algorithm for a problem and estimate its time and space complexities.
- Possess the skill for representing knowledge using the appropriate technique for a given problem.
- 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.**UNIT-II:****Problem Solving by Search-II:** 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.**UNIT-III:****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.**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.**UNIT-IV:****Knowledge Representation:** Ontological Engineering, Categories and Objects, Events. Mental Events and Mental Objects, Reasoning Systems for Categories, Reasoning with Default Information.**Planning - Classical Planning:** Definition of Classical Planning, Algorithms for Planning with State-Space Search, Planning Graphs, other Classical Planning Approaches, Analysis of Planning approaches.

**UNIT-V:**

**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.

**TEXT BOOK:**

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

**REFERENCES:**

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

**INFORMATION RETRIEVAL SYSTEMS (Professional Elective - I)****MCA II Year I Sem.**

L	T	P	C
3	0	0	3

**Prerequisites:**

1. Data Structures

**Course Objectives:**

- To learn the important concepts and algorithms in IRS
- To understand the data/file structures that are necessary to design, and implement information retrieval (IR) systems.

**Course Outcomes:**

- Ability to apply IR principles to locate relevant information large collections of data
- Ability to design different document clustering algorithms
- Implement retrieval systems for web search tasks.
- Design an Information Retrieval System for web search tasks.

**UNIT - I**

Introduction to Information Retrieval Systems: Definition of Information Retrieval System, Objectives of Information Retrieval Systems, Functional Overview, Relationship to Database Management Systems, Digital Libraries and Data Warehouses

Information Retrieval System Capabilities: Search Capabilities, Browse Capabilities, Miscellaneous Capabilities

**UNIT - II**

Cataloging and Indexing: History and Objectives of Indexing, Indexing Process, Automatic Indexing, Information Extraction

Data Structure: Introduction to Data Structure, Stemming Algorithms, Inverted File Structure, N-Gram Data Structures, PAT Data Structure, Signature File Structure, Hypertext and XML Data Structures, Hidden Markov Models

**UNIT - III**

Automatic Indexing: Classes of Automatic Indexing, Statistical Indexing, Natural Language, Concept Indexing, Hypertext Linkages

Document and Term Clustering: Introduction to Clustering, Thesaurus Generation, Item Clustering, Hierarchy of Clusters

**UNIT - IV**

User Search Techniques: Search Statements and Binding, Similarity Measures and Ranking, Relevance Feedback, Selective Dissemination of Information Search, Weighted Searches of Boolean Systems, Searching the INTERNET and Hypertext

Information Visualization: Introduction to Information Visualization, Cognition and Perception, Information Visualization Technologies

**UNIT - V**

Text Search Algorithms: Introduction to Text Search Techniques, Software Text Search Algorithms, Hardware Text Search Systems

Multimedia Information Retrieval: Spoken Language Audio Retrieval, Non-Speech Audio Retrieval, Graph Retrieval, Imagery Retrieval, Video Retrieval

**TEXT BOOK:**

1. Information Storage and Retrieval Systems – Theory and Implementation, Second Edition, Gerald J. Kowalski, Mark T. Maybury, Springer

**REFERENCES:**

1. Frakes, W.B., Ricardo Baeza-Yates: Information Retrieval Data Structures and Algorithms, Prentice Hall, 1992.
2. Information Storage & Retrieval By Robert Korfhage – John Wiley & Sons.
3. Modern Information Retrieval By Yates and Neto Pearson Education.

**AD-HOC AND SENSOR NETWORKS (Professional Elective - I)**

MCA II Year I Sem.

L	T	P	C
3	0	0	3

**Prerequisites**

- Computer Networks
- Distributed Systems
- Mobile Computing

**Course Objectives**

- To understand the concepts of sensor networks
- To understand the MAC and transport protocols for adhoc networks
- To understand the security of sensor networks
- To understand the applications of adhoc and sensor networks

**Course Outcomes**

- Understand challenges of MANETs and routing in MANETs in ad hoc and wireless sensor networks (ASN)
- Analyze data transmission and geocasting in ad hoc and wireless sensor networks (ASN)
- Understand basics of Wireless, Sensors and Lower Layer Issues and Upper Layer Issues of WSN

**UNIT - I****Introduction to Ad Hoc Networks**

Characteristics of MANETs, Applications of MANETs and Challenges of MANETs.

**Routing in MANETs**

Criteria for classification, Taxonomy of MANET routing algorithms, *Topology-based* routing algorithms-Proactive: DSDV, WRP; Reactive: DSR, AODV, TORA; Hybrid: ZRP; *Position-based* routing algorithms-Location Services-DREAM, Quorum-based, GLS; Forwarding **Strategies** Greedy Packet, Restricted Directional Flooding-DREAM, LAR; Other routing algorithms-QoS Routing, CEDAR.

**UNIT - II****Data Transmission**

Broadcast Storm Problem, Rebroadcasting Schemes-Simple-flooding, Probability-based Methods, Area-based Methods, Neighbour Knowledge-based: SBA, Multipoint Relaying, AHBP. Multicasting: Tree-based: AMRIS, MAODV; Mesh-based: ODMRP, CAMP; Hybrid: AMRoute, MCEDAR and Geocasting: Data-transmission Oriented-LBM; Route Creation Oriented-GeoTORA, MGR.

**UNIT - III****Geocasting**

Data-transmission Oriented-LBM; Route Creation Oriented-GeoTORA, MGR. TCP over Ad Hoc TCP protocol overview, TCP and MANETs, Solutions for TCP over Ad hoc

**UNIT - IV****Basics of Wireless, Sensors and Lower Layer Issues**

Applications, Classification of sensor networks, Architecture of sensor network, Physical layer, MAC layer, Link layer, Routing Layer.

**UNIT - V****Upper Layer Issues of WSN**

Transport layer, High-level application layer support, Adapting to the inherent dynamic nature of WSNs, Sensor Networks and mobile robots.

**TEXT BOOKS:**

1. Ad Hoc and Sensor Networks – Theory and Applications, *Carlos Corderio Dharma P. Aggarwal*, World

Scientific Publications, March 2006, ISBN – 981-256-681-3

2. Wireless Sensor Networks: An Information Processing Approach, Feng Zhao, Leonidas Guibas, Elsevier Science, ISBN – 978-1-55860-914-3 (Morgan Kauffman)

**REFERENCES:**

1. C. Siva Ram Murthy, B.S. Manoj Ad Hoc Wireless Networks: Architectures and Protocols
2. Taieb Znati Kazem Sohraby, Daniel Minoli, Wireless Sensor Networks: Technology, Protocols and Applications, Wiley

**CYBER SECURITY (Professional Elective - II)**

MCA II Year I Sem.

L	T	P	C
3	0	0	3

**Course objectives:**

- To understand various types of cyber-attacks and cyber-crimes
- To learn threats and risks within context of the cyber security
- To have an overview of the cyber laws & concepts of cyber forensics
- To study the defensive techniques against these attacks

**Course Outcomes:**

- Analyze and evaluate the cyber security needs of an organization.
- Understand Cyber Security Regulations and Roles of International Law
- Design and develop a security architecture for an organization.
- Understand fundamental concepts of data privacy attacks

**UNIT-I**

**Introduction to Cyber Security:** Basic Cyber Security Concepts, layers of security, Vulnerability, threat, Harmful acts, Internet Governance – Challenges and Constraints, Computer Criminals, CIA Triad, Assets and Threat, motive of attackers, active attacks, passive attacks, Software attacks, hardware attacks, Cyber Threats-Cyber Warfare, Cyber Crime, Cyber terrorism, Cyber Espionage, etc., Comprehensive Cyber Security Policy.

**UNIT-II**

**Cyberspace and the Law & Cyber Forensics:** Introduction, Cyber Security Regulations, Roles of International Law. The INDIAN Cyberspace, National Cyber Security Policy. Introduction, Historical background of Cyber forensics, Digital Forensics Science, The Need for Computer Forensics, Cyber Forensics and Digital evidence, Forensics Analysis of Email, Digital Forensics Lifecycle, Forensics Investigation, Challenges in Computer Forensics

**UNIT-III**

**Cybercrime: Mobile and Wireless Devices:** Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication service Security, Attacks on Mobile/Cell Phones, Organizational security Policies and Measures in Mobile Computing Era, Laptops.

**UNIT-IV**

**Cyber Security: Organizational Implications:** Introduction, cost of cybercrimes and IPR issues, web threats for organizations, security and privacy implications, social media marketing: security risks and perils for organizations, social computing and the associated challenges for organizations

**UNIT-V**

**Privacy Issues:** Basic Data Privacy Concepts: Fundamental Concepts, Data Privacy Attacks, Data linking and profiling, privacy policies and their specifications, privacy policy languages, privacy in different domains- medical, financial, etc

**Cybercrime: Examples and Mini-Cases**

**Examples:** Official Website of Maharashtra Government Hacked, Indian Banks Lose Millions of Rupees, Parliament Attack, Pune City Police Bust Nigerian Racket, e-mail spoofing instances. **Mini-Cases:** The Indian Case of online Gambling, An Indian Case of Intellectual Property Crime, Financial Frauds in Cyber Domain.

**TEXT BOOKS:**

1. Nina Godbole and Sunit Belpure, Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley
2. B.B. Gupta, D.P. Agrawal, Haoxiang Wang, Computer and Cyber Security: Principles, Algorithm, Applications, and Perspectives, CRC Press, ISBN 9780815371335,2018.

**REFERENCES:**

1. Cyber Security Essentials, James Graham, Richard Howard and Ryan Otson, CRC Press.
2. Introduction to Cyber Security, Chwan-Hwa(john) Wu,J. David Irwin, CRC Press T & F Group.



**MOBILE COMPUTING (Professional Elective - II)**

MCA II Year I Sem.

L	T	P	C
3	0	0	3

**Prerequisites:**

- Computer Networks, Distributed Systems / Distributed Operating Systems

**Course Objectives:**

- To make the student understand the concept of mobile computing paradigm, its novel applications and limitations, the typical mobile networking infrastructure through a popular GSM protocol, the issues and solutions of various layers of mobile networks.

**Course Outcomes:**

- Understand the concept of mobile computing paradigm, its novel applications and limitations.
- Analyze and develop new mobile applications
- Understand the protocols and platforms related to mobile environment
- Classify data delivery mechanisms

**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, CSHSD, DECT.

**UNIT -II**

**(Wireless) Medium Access Control (MAC):** Motivation for a specialized MAC (Hidden and exposed terminals, Near and far 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, Query processing, Data Recovery Process & QoS Issues.

**UNIT -IV**

**Data Dissemination and Synchronization:** Communications Asymmetry, Classification of Data Delivery Mechanisms, Data Dissemination, Broadcast Models, Selective Tuning and Indexing Methods, Data Synchronization – Introduction, Software, and Protocols

**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.

**TEXT BOOKS:**

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

**REFERENCE:**

1. Asoke K Talukder, Hasan Ahmed, Roopa Yavagal Mobile Computing: Technology, Applications and Service Creation, McGraw Hill Education

**SOFTWARE TESTING METHODOLOGIES (Professional Elective - II)****MCA II Year I Sem.**

L	T	P	C
3	0	0	3

**Prerequisites**

- A course on “Software Engineering”.

**Course Objectives:**

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

**Course Outcomes:**

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

**UNIT - I**

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

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

**UNIT - II**

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

**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.

**ENTERPRISE CLOUD CONCEPTS LAB****MCA II Year I Sem.****L T P C**  
**0 0 3 1.5****Course Objectives:**

- Knowledge on significance of cloud computing and its fundamental concepts and models.

**Course Outcomes:**

- Understand importance of cloud architecture
- Illustrating the fundamental concepts of cloud security
- Analyze various cloud computing mechanisms
- Understanding the architecture and working of cloud computing.

**List of Experiments:**

1. Install Virtualbox/VMware Workstation with different flavors of linux or windows OS on top of windows7 or 8.
2. Install a C compiler in the virtual machine created using virtual box and execute Simple Programs
3. Install Google App Engine. Create a hello world app and other simple web applications using python/java.
4. Find a procedure to transfer the files from one virtual machine to another virtual machine.
5. Find a procedure to launch virtual machine using trystack (Online Openstack Demo Version)
6. Install Hadoop single node cluster and run simple applications like word count.

**E-Resources:**

1. <https://www.iitk.ac.in/nt/faq/vbox.htm>
2. <https://www.google.com/urlsa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwjqrNG0za73AhXZt1YBHZ21DWEQFnoECAMQAQ&url=http%3A%2F%2Fwww.cs.columbia.edu%2F~sedwards%2Fclasses%2F2015%2F1102-fall%2Flinuxvm.pdf&usg=AOvVaw3xZPuF5xVgk-AQnBRsTtHz>
3. <https://www.cloudsimtutorials.online/cloudsim/>
4. <https://edwardsamuel.wordpress.com/2014/10/25/tutorial-creating-openstack-instance-in-trystack/>
5. <https://www.edureka.co/blog/install-hadoop-single-node-hadoop-cluster>

**FULL STACK DEVELOPMENT LAB****MCA II Year I Sem.**

L	T	P	C
0	0	3	1.5

**Course Objectives:** The students should be able:

- To implement Forms, inputs and Services using AngularJS
- To develop a simple web application using Nodejs; Angular JS and Express
- To implement data models using MongoDB

**Course Outcomes:**

- Develop a fully functioning website and deploy on a web server.
- Gain Knowledge about the front end and back end Tools
- Find and use code packages based on their documentation to produce working results in a project.
- Create web pages that function using external data.

**List of Experiments:**

1. Develop a Form and validate using AngularJS
2. Create and implement modules and controllers in AngularJS
3. Implement Error Handling in AngularJS
4. Create and implement Custom directives
5. Create a simple web application using Express, Node JS and Angular JS
6. Implement CRUD operations on MongoDB
7. Create a react application for the student management system having registration, login, contact, about pages and implement routing to navigate through these pages.
8. Create a service in react that fetches the weather information from openweathermap.org and the display the current and historical weather information using graphical representation using chart.js
9. Create a TODO application in react with necessary components and deploy it into github.
10. A. Develop an express web application that can interact with REST API to perform CRUD operations on student data. (Use Postman)  
B. For the above application create authorized end points using JWT (JSON Web Token).

**MACHINE LEARNING LAB****MCA II Year I Sem.**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>1</b>	<b>2</b>	<b>2</b>

**Course Objective:**

- The objective of this lab is to get an overview of the various machine learning
- Techniques and can demonstrate them using python.

**Course Outcomes:**

- Understand modern notions in predictive data analysis
- Select data, model selection, model complexity and identify the trends
- Understand a range of machine learning algorithms along with their strengths and weaknesses
- Build predictive models from data and analyze their performance

**List of Experiments**

1. Write a python program to compute Central Tendency Measures: Mean, Median, Mode Measure of Dispersion: Variance, Standard Deviation
2. Study of Python Basic Libraries such as Statistics, Math, Numpy and Scipy
3. Study of Python Libraries for ML application such as Pandas and Matplotlib
4. Write a Python program to implement Simple Linear Regression
5. Implementation of Multiple Linear Regression for House Price Prediction using sklearn
6. Implementation of Decision tree using sklearn and its parameter tuning
7. Implementation of KNN using sklearn
8. Implementation of Logistic Regression using sklearn
9. Implementation of K-Means Clustering
10. Performance analysis of Classification Algorithms on a specific dataset (Mini Project)

**TEXT BOOK:**

1. Machine Learning – Tom M. Mitchell, - MGH

**REFERENCE:**

1. Machine Learning: An Algorithmic Perspective, Stephen Marshland, Taylor & Francis

**INTERNET OF THINGS (Professional Elective - III)****MCA II Year II Sem.**

L	T	P	C
3	0	0	3

**Course Objectives:**

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

**Course Outcomes:**

- Interpret the impact and challenges posed by IoT networks leading to new architectural models.
- Compare and contrast the deployment of smart objects and the technologies to connect them to the network.
- Appraise the role of IoT protocols for efficient network communication.
- Elaborate the need for Data Analytics and Security in IoT.
- 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.

**MOBILE APPLICATION DEVELOPMENT (Professional Elective - III)****MCA II Year II Sem.**

L	T	P	C
3	0	0	3

**Prerequisites**

- Acquaintance with JAVA programming
- A Course on DBMS

**Course Objectives**

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

**Course Outcomes**

- Students understand the working of Android OS Practically.
- Student will be able to develop Android user interfaces
- Students 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 UNIT - s 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)

**TEXT BOOKS:**

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

**REFERENCE:**

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



**BIG DATA ANALYTICS (Professional Elective - III)**

MCA II Year II Sem.

L	T	P	C
3	0	0	3

**Course Objectives**

- The purpose of this course is to provide the students with the knowledge of Big data Analytics principles and techniques.
- This course is also designed to give an exposure of the frontiers of Big data Analytics

**Courses Outcomes**

- Ability to explain the foundations, definitions, and challenges of Big Data and various Analytical tools.
- Ability to program using HADOOP and Map reduce, NOSQL
- Ability to understand the importance of Big Data in Social Media and Mining.

**UNIT - I****Getting an Overview of Big Data**

What is Big Data? History of Data Management – Evolution of Big Data, Structuring Big Data, Elements of Big Data, Big Data Analytics, Careers in Big Data, Future of Big Data

**Technologies for Handling Big Data**

Distributed and Parallel Computing for Big Data, Introducing Hadoop, Cloud Computing and Big Data, In-Memory Computing Technology for Big Data.

**UNIT -II****Understanding Hadoop Ecosystem**

Hadoop Ecosystem, Hadoop Distributed File System, MapReduce, Hadoop YARN, Hbase, Hive, Pig and Pig Latin, Sqoop, ZooKeeper, Flume, Oozie

**Understanding MapReduce Fundamentals and HBase**

The MapReduce Framework, Techniques to Optimize MapReduce Jobs, Uses of MapReduce, Role of HBase in Big Data Processing

**UNIT -III****Understanding Analytics and Big Data**

Comparing Reporting and Analysis, Types of Analytics, Points to Consider during Analysis, Developing an Analytic Team, Understanding Text Analytics

**Analytical Approaches and Tools to Analyze Data:** Analytical Approaches, History of Analytical Tools  
Introduction to Popular Analytical Tools, Comparing Various Analytical Tools, Installing R

**UNIT -IV****Data Visualization-I**

Introducing Data Visualization, Techniques Used for Visual Data Representation, Types of Data Visualization, Applications of Data Visualization, Visualizing Big Data, Tools Used in Data Visualization, Tableau Products

**Data Visualization with Tableau (Data Visualization-II):** Introduction to Tableau Software, Tableau Desktop Workspace, Data Analytics in Tableau Public, Using Visual Controls in Tableau Public

**UNIT -V****Social Media Analytics and Text Mining**

Introducing Social Media, Introducing Key Elements of Social Media, Introducing Text Mining, Understanding Text Mining Process, Sentiment Analysis, Performing Social Media Analytics and Opinion Mining on Tweets

**Mobile Analytics:** Introducing Mobile Analytics, Introducing Mobile Analytics Tools, Performing Mobile Analytics, Challenges of Mobile Analytics

**TEXT BOOKS:**

1. Big data, blackbook, dreamtech press, 2015
2. Big Data Analytics, SeemaAcharya, Subhashini Chellappan, Wiley 2015.
3. Simon Walkowiak, Big Data Analytics with R, Packt Publishing, ISBN: 9781786466457

**REFERENCES:**

1. Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Business, Michael Minelli, Michehe Chambers, 1<sup>st</sup> Edition, AmbigaDhiraj, Wiley CIO Series, 2013.
2. Hadoop: The Definitive Guide, Tom White, 3<sup>rd</sup> Edition, O'Reilly Media, 2012.
3. Big Data Analytics: Disruptive Technologies for Changing the Game, Arvind Sathi, 1<sup>st</sup> Edition, IBM Corporation, 2012.

**E-COMMERCE (Open Elective - I)**

MCA II Year II Sem.

L	T	P	C
3	0	0	3

**Course Objectives:**

- Identify the major categories and trends of e-commerce applications.
- Identify the essential processes of an e-commerce system.
- Identify several factors and web store requirements needed to succeed in e-commerce.
- Discuss the benefits and trade-offs of various e-commerce clicks and bricks alternatives.
- Understand the main technologies behind e-commerce systems and how these technologies interact.
- Discuss the various marketing strategies for an online business.
- Define various electronic payment types and associated security risks and the ways to protect against them.

**Course Outcomes:**

- Ability to identify the business relationships between the organizations and their customers.
- Ability to perform various transactions like payment, data transfer and etc.

**UNIT-I:**

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

**UNIT-II:**

**Electronic payment systems:** Digital Token-Based, Smart Cards, Credit Cards, Risks in Electronic Payment systems. Inter Organizational Commerce - EDI, EDI Implementation, Value added networks. Intra Organizational Commerce - work Flow, Automation Customization and internal Commerce, Supply chain Management.

**UNIT-III:**

**Corporate Digital Library:** Document Library, digital Document types, corporate Data Warehouses. Advertising and Marketing - Information based marketing, Advertising on Internet, on-line marketing process, market research. Consumer Search and Resource Discovery - Information search and Retrieval, Commerce Catalogues, Information Filtering. Multimedia - key multimedia concepts, Digital Video and electronic Commerce, Desktop video processing's, Desktop video conferencing

**UNIT-IV:**

**Web Marketing Strategies:** Communicating with Different Market Segments, Beyond Market Segmentation: Customer Behavior and Relationship Intensity, Advertising on the Web, EMail Marketing, Search Engine Positioning and Domain Names, Selling to Businesses Online, Electronic Data Interchange, Supply Chain Management Using Internet Technologies, Electronic Marketplaces and Portals

**UNIT-V:**

**E-Business Revenue Models:** Revenue Models for Online Business, Changing Strategies: Revenue Models in Transition, Revenue Strategy Issues for Online Businesses, Creating an Effective Business Presence Online, Web Site Usability, Virtual Communities, Mobile Commerce, Online Auctions

**TEXT BOOKS:**

1. Frontiers of electronic commerce – Kalakata, Whinston, Pearson. (UNITS 1, 2, 3)
2. E-Business by Gary P. Schneider, - Cengage India Learning (UNITS 4, 5).

**REFERENCES:**

1. E-Commerce fundamentals and applications Hendry Chan, Raymond Lee, Tharam Dillon, Elizabeth Chang, John Wiley.
2. E-Commerce, S. Jaiswal – Galgotia.
3. E-Commerce, Efrain Turbon, Jae Lee, David King, H. Michael Chang.
4. Electronic Commerce – Gary P. Schneider – Thomson.

**CYBER LAWS & ETHICS (Open Elective - I)****MCA II Year II Sem.**

L	T	P	C
3	0	0	3

**Course Objectives:**

- To make the students understand the types of roles they are expected to play in the society as practitioners of the civil engineering profession
- To develop some ideas of the legal and practical aspects of their profession.

**Course Outcomes:**

- Understand the importance of professional practice, Law and Ethics in their personal lives and professional careers.
- Learn the rights and responsibilities as an employee, team member and a global citizen
- Understand the information processing and secure program administration
- Understand the fundamentals of Organizational and Human security standards

**UNIT-I**

Introduction to Computer Security: Definition, Threats to security, Government requirements, Information Protection and Access Controls, Computer security efforts, Standards, Computer Security mandates and legislation, Privacy considerations, International security activity.

**UNIT-II**

Secure System Planning and administration, Introduction to the orange book, Security policy requirements, accountability, assurance and documentation requirements, Network Security, The Red book and Government network evaluations.

**UNIT-III**

Information security policies and procedures: Corporate policies- Tier 1, Tier 2 and Tier3 policies - process management-planning and preparation-developing policies-asset classification policy- developing standards.

**UNIT-IV**

Information security: fundamentals-Employee responsibilities- information classification- Information handling- Tools of information security- Information processing-secure program administration.

**UNIT-V**

Organizational and Human Security: Adoption of Information Security Management Standards, Human Factors in Security- Role of information security professionals.

**TEXT BOOKS:**

1. Debby Russell and Sr. G. T Gangemi, "Computer Security Basics (Paperback)", 2<sup>nd</sup> Edition, O' Reilly Media, 2006.
2. Thomas R. Peltier, "Information Security policies and procedures: A Practitioner's Reference", 2<sup>nd</sup> Edition Prentice Hall, 2004.

**REFERENCES:**

1. Kenneth J. Knapp, "Cyber Security and Global Information Assurance: Threat Analysis and Response Solutions", IGI Global, 2009.
2. Thomas R Peltier, Justin Peltier and John blackley," Information Security Fundamentals", 2<sup>nd</sup> Edition, Prentice Hall, 1996
3. Jonathan Rosenoer, "Cyber law: the Law of the Internet", Springer-verlag, 1997 James Graham, "Cyber Security Essentials" Averbach Publication T & F Group.

**MANAGEMENT INFORMATION SYSTEMS (Open Elective - I)****MCA II Year II Sem.**

L	T	P	C
3	0	0	3

**Prerequisites**----NIL----**Course Objectives:**

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

**Course Outcomes:**

- To understand different Types of Information Systems
- 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.

**TEXT BOOK:**

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.

**ENTREPRENEURSHIP (Open Elective - I)**

MCA II Year II Sem.

L	T	P	C
3	0	0	3

**Course Objectives:**

- The aim of this course is to have a comprehensive perspective of inclusive learning, ability to learn and implement the Fundamentals of Entrepreneurship

**Course Outcomes:**

- It enables students to learn the basics of Entrepreneurship and entrepreneurial development which will help them to provide vision for their own Start-up.

**UNIT-I:**

**Entrepreneurial Perspectives:** Introduction to Entrepreneurship – Evolution - Concept of Entrepreneurship - Types of Entrepreneurs - Entrepreneurial Competencies, Capacity Building for Entrepreneurs. Entrepreneurial Training Methods - Entrepreneurial Motivations - Models for Entrepreneurial Development - The process of Entrepreneurial Development.

**UNIT-II:**

**New Venture Creation:** Introduction, Mobility of Entrepreneurs, Models for Opportunity Evaluation; Business plans – Purpose, Contents, Presenting Business Plan, Procedure for setting up Enterprises, Central level - Startup and State level - T Hub, Other Institutions initiatives.

**UNIT-III:**

**Management of MSMEs and Sick Enterprises:** Challenges of MSMEs, Preventing Sickness in Enterprises – Specific Management Problems; Industrial Sickness; Industrial Sickness in India – Symptoms, process and Rehabilitation of Sick Units.

**UNIT-IV:**

**Managing Marketing and Growth of Enterprises:** Essential Marketing Mix of Services, Key Success Factors in Service Marketing, Cost and Pricing, Branding, New Techniques in Marketing, International Trade.

**UNIT-V:**

**Strategic perspectives in Entrepreneurship:** Strategic Growth in Entrepreneurship, The Valuation Challenge in Entrepreneurship, The Final Harvest of New Ventures, Technology, Business Incubation, India way – Entrepreneurship; Women Entrepreneurs – Strategies to develop Women Entrepreneurs, Institutions supporting Women Entrepreneurship in India.

**TEXT BOOKS:**

1. Entrepreneurship Development and Small Business Enterprises, Poornima M. Charantimath, 2nd edition, Pearson, 2014.
2. Entrepreneurship, a South – Asian Perspective, D.F. Kuratko and T.V.Rao, 3rd edition, Cengage, 2012.

**REFERENCES:**

1. Entrepreneurship, Arya Kumar, 4th edition, Pearson 2015.
2. The Dynamics of Entrepreneurial Development and Management, Vasant Desai, Himalaya Publishing House, 2015.