Course Title	**DISCRETE MATHEMATICAL STRUCTURES AND PROBABILITY**	Semester	III
Course Code	MVJ19MCS31/IS31	**CIE**	50
Total No. of Contact Hours	60 L : T : P :: 40 : 0 : 20	**SEE**	50
No. of Contact Hours/week	4	**Total**	100
Credits	3	**Exam. Duration**	3 Hours

Course objective is to:

- Prepare for a background in abstraction, notation, and critical thinking for the mathematics most directly related to computer science.
- Understand and apply logic, relations, functions, basic set theory, countability and counting arguments, proof techniques,
- Understand and apply mathematical induction, combinatorics, discrete probability, recursion, sequence and recurrence, elementary number theory
- Understand and apply graph theory and mathematical proof techniques.

Module-1	**RBT Level** L1,L2 & L3	**10 Hrs.**

Properties of the Integers: Mathematical Induction, The Well Ordering Principle − Mathematical Induction, Recursive Definitions. Principles of Counting. Fundamental Principles of Counting: The Rules of Sum and Product, Permutations, Combinations − The Binomial Theorem, Combinations with Repetition

Module-2	**RBT Level** L1,L2 & L3	**10 Hrs.**

The Principle of Inclusion and Exclusion: The Principle of Inclusion and Exclusion, Generalizations of the Principle, Derangements − Nothing is in its Right Place, Rook Polynomials. Recurrence Relations: First Order Linear Recurrence Relation, The Second Order Linear Homogeneous Recurrence Relation with Constant Coefficients,

Module-3	**RBT Level** L1,L2 & L3	**10 Hrs.**

Probability Theory: Basic terminology, Definition of probability, Probability and set notations, Addition law of probability, independent events, conditional probability, multiplication law of probability, Baye's theorem.

Module-4	**RBT Level** L1,L2 & L3	**10 Hrs.**

Probability Distributions: Random variables (discrete and continuous), probability mass/density functions. Binomial distribution, Poisson distribution. Exponential and normal distributions, problems.Joint probability distribution: Joint Probability distribution for two discrete random

variables, expectation, covariance, correlation coefficient.		
Module-5	**RBT Level** L1,L2 & L3	**10 Hrs.**
Sampling Theory: Sampling, Sampling distributions, standard error, test of hypothesis for means and proportions, confidence limits for means, student's t-distribution, Chi-square distribution as a test of goodness of fit. Stochastic process:Stochastic processes, probability vector, stochastic matrices, fixed points, regular stochastic matrices, Markov chains, higher transition probability simple problems.		
Course outcomes:		
CO1	Demonstrate the application of discrete structures in different fields of computer Science.	
CO2	Solve problems using recurrence relations and generating functions.	
CO3	Determine the nature of the events and hence calculate the appropriate probabilities of the events	
CO4	Develop probability distribution of discrete, continuous random variables and joint probability distribution occurring in digital signal processing, information theory and design engineering.	
CO5	Demonstrate testing of hypothesis of sampling distributions and illustrate examples of Markov chains related to discrete parameter stochastic process.	

Reference Books:	
1.	B.S. Grewal, "Higher Engineering Mathematics" Khanna Publishers, 43rd Edition, 2013.
2.	Ralph P. Grimaldi: Discrete and Combinatorial Mathematics, 5th Edition, Pearson Education. 2004.
3	Ramana B. V., "Higher Engineering Mathematics", Tata Mc Graw-Hill, 2006.
4	Bali N. P. & Manish Goyal, "A text book of Engineering Mathematics", Laxmi Publications, 8th Edition
	M. K. Jain, S. R. K. Iyengar and R. K. Jain "Numerical Methods: For Scientific and Engineering Computation", New Age International Publications, 6th Edition, 2012, ISBN: 9788122433234.
6	Basavaraj S Anami and Venakanna S Madalli: Discrete Mathematics – A Concept based approach, Universities Press, 2016
7	Kenneth H. Rosen: Discrete Mathematics and its Applications, 6th Edition, McGraw Hill, 2007

CO-PO Mapping												
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12

CO1	3	3	0	3	0	0	0	0	0	0	1	1
CO2	2	3	0	3	0	0	0	0	0	0	1	1
CO3	2	3	0	3	0	0	0	0	0	0	1	1
CO4	3	3	0	3	0	0	0	0	0	0	1	1
CO5	3	3	0	3	0	0	0	0	0	0	1	1

High-3, Medium-2, Low-1

Course Title	**DATA STRUCTURES AND APPLICATIONS**	Semester	III
Course Code	MVJ19CS32/IS32	**CIE**	50
Total No. of Contact Hours	60 L : T : P :: 40 : 0 : 20	**SEE**	50
No. of Contact Hours/week	6	**Total**	100
Credits	4	**Exam. Duration**	3 Hours

Course objective:

The Students will be able to

- Identify the importance of data structures & memory allocation.
- Perform operations on stacks and queues and its applications.
- Apply the operations of linked list, Trees & Graphs in various applications.
- Apply searching and sorting operations in real time applications.

Module-1	**RBT Level** L1, L2, L3	**11 Hrs.**

Introduction: Data Structures, Classifications (Primitive & Non Primitive), Data structure Operations, Review of Arrays, Structures, Self-Referential Structures. Pointers and Dynamic Memory Allocation Functions. Representation of Linear Arrays in Memory, Dynamically allocated arrays. Abstract Data Type**, Array Operations:** Traversing, inserting, deleting, searching, and sorting**, Array ADT** Multidimensional Arrays, Polynomials and Sparse Matrices. **Strings:** Basic Terminology, Storing, Operations and Pattern Matching algorithms. Programming Examples.

Laboratory Sessions/ Experimental learning:

1. Create an array of structure which has the following members Student name, Student USN, Marks1, Marks2, Marks3. Allocate memory to store 5 students details initially. When a new student details need to be entered or to be deleted in this array, dynamically change the array size. Write a program to implement this scenario and display the result.
2. Find the bug for the following code and then Debug it

```
int minval(int *A, int n) {
int currmin;
for (int i=0; i<n; i++)
  if (A[i] < currmin)
    currmin = A[i];
 return currmin;
}
```

3. Compile the following code and debug it.

```
#include <stdio.h>
#include <string.h>
struct student
{
 int id;
 char name[30];
 float percentage;
};
int main()
{
    int i;
    struct student record1 = {1, "Raju", 90.5};
    struct student *ptr;
       printf("Records of STUDENT1: \n");
       printf("  Id is: %d \n", ptr->id);
       printf("  Name is: %s \n", ptr->name);
       printf("  Percentage is: %f \n\n", ptr->percentage);
     return 0;
}
```

Real Time Applications: System memory allocation

Video link / Additional online information (related to module if any):

1. https://nptel.ac.in/courses/106106130/
2. https://nptel.ac.in/courses/106105085/
3. https://nptel.ac.in/courses/106/106/106106127/
4. https://www.coursera.org/lecture/data-structures/arrays-OsBSF

Module-2	**RBT Level** L1, L2, L3	**10 Hrs.**

Stacks: Definition, Stack Operations, Stack ADT, Array Representation of Stacks, Stacks using Dynamic Arrays, Stack Applications: Polish notation, Infix to postfix conversion, evaluation of postfix expression. **Recursion -** GCD, Tower of Hanoi. **Queues:** Definition, Array Representation, Queue Operations, Queue ADT, Circular Queues, Circular queues using Dynamic arrays, Dequeues,

Priority Queues. Programming Examples.

Laboratory Sessions/ Experimental learning:

Design, Develop and Implement a menu driven Program in C for the following operations on DEQUEUE of Integers (Array Implementation of Queue with maximum size MAX)

a. Insert an Element on to DEQUEUE

b. Delete an Element from DEQUEUE

c. Demonstrate Overflow and Underflow situations on DEQUEUE

d. Display the status of DEQUEUE

e. Exit Support the program with appropriate functions for each of the above operations

Real Time Applications: Game applications, Ticket booking applications (Eg: Train, restaurant etc)
Video link / Additional online information (related to module if any):

1.https://nptel.ac.in/courses/106106130/

2. https://nptel.ac.in/courses/106102064/

3. https://nptel.ac.in/courses/106105085/

4. https://nptel.ac.in/courses/106/106/106106127/

Module-3	**RBT Level** L1, L2, L3	**13 Hrs.**

Linked Lists: Definition, Representation of linked lists in Memory, Memory allocation; Garbage Collection. Linked list operations: Traversing, Searching, Insertion, and Deletion. Doubly Linked lists, Circular linked lists, and header linked lists. Linked Stacks and Queues. Applications of Linked lists – Polynomials. Programming Examples

Hashing: Hash Table organizations, Hashing Functions, Static and Dynamic Hashing.

Laboratory Sessions/ Experimental learning:

1.Design, Develop and Implement a Program in C for the following operations on Singly Circular Linked List (SCLL) with header nodes a. Represent and Evaluate a Polynomial P(x,y,z) = 6x2 y 2 z-4yz5 +3x3 yz+2xy5 z-2xyz3 b. Find the sum of two polynomials POLY1(x,y,z) and POLY2(x,y,z) and store the result in POLYSUM(x,y,z) Support the program with appropriate functions for each of the above operations

2. Debug the following code and explain the process

```
//Insert a value into an ordered linked list
void insert(lnode*& curr, int val) {
  if (curr == NULL)
```

```
    curr = new lnode(val, NULL);
  else if (lnode->val > val)
    curr = new lnode(val, curr->next);
  else {
    curr = curr->next;
    insert(curr, val);
  }
}
```

Real Time Applications: Music Player, Image Viewer, Web browser, Process Management, Mechanical field

Video link / Additional online information (related to module if any):

1. https://nptel.ac.in/courses/106106130/
2. https://nptel.ac.in/courses/106102064/
3. https://nptel.ac.in/courses/106105085/

Module-4	**RBT Level** L1, L2, L3	**13 Hrs.**

Trees: Terminology, Binary Trees, Properties of Binary trees, Array and linked Representation of Binary Trees, Binary Tree Traversals - Inorder, postorder, preorder; Additional Binary tree operations. Threaded binary trees, Binary Search Trees – Definition, Insertion, Deletion, Traversal, Searching, Application of Trees-Evaluation of Expression, AVL Trees, Splay Trees, B-Tree, Programming Examples

Laboratory Sessions/ Experimental learning:

Design, Develop and Implement a menu driven Program in C for the following operations on AVL Trees

i) Construct an AVL tree by inserting the following elements in the given order.

63, 9, 19, 27, 18, 108, 99, 81.

ii)searching for a node

iii)Deleting a node

Real Time Applications: Indexing in databases, Programming Languages, Computer chess games, Computer file system, Undo function in text editor, representing city region telehone network etc.

Video link:

1.https://nptel.ac.in/courses/106102064/

2. http://www.digimat.in/nptel/courses/video/106106127/L50.html

3. https://www.youtube.com/watch?v=ffgg_zmbaxw		
Module-5	**RBT Level** L1, L2, L3	**13 Hrs.**
Graphs: Definitions, Terminologies, Matrix and Adjacency List Representation of Graphs, Elementary Graph operations, Traversal methods: Breadth First Search and Depth First Search, Topological Sort. **Sorting and Searching:** Quick sort, Insertion Sort, Radix sort, Merge Sort, Address Calculation Sort. **Laboratory Sessions/ Experimental learning:** Sort a given set of elements using the sorting Method which divides input array in two halves, calls itself for the two halves and then merges the two sorted halves" and determine the time required to sort the elements. Repeat the experiment for different values of n, thenumber of elements in the list to be sorted and plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator. Real Time Applications: Graph Theory, E-Commerce websites, Google Maps, Facebook Video link: https://www.youtube.com/watch?v=hk5rQs7TQ7E&feature=youtu.be https://nptel.ac.in/courses/106/102/106102064/		
Course outcomes:		
CO1	Identify the necessity of data structure and its storage process.	
CO2	Analyse the various operations performed on stack and queues for different applications.	
CO3	Perform various operations on linked list for different applications**.**	
CO4	Learn Trees and its applications.	
CO5	Analyse the concepts of Graphs, searching, sorting & hashing in real time.	

Reference Books:	
1.	Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Ed, Universities Press, 2014.
2.	Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.
3.	Reema Thareja, Data Structures using C, 3rd Ed, Oxford press, 2012.
4.	Mark Allen Weiss, ―Data Structures and Algorithm Analysis in C‖, 2nd Edition, Pearson Education,1997

5.	Gilberg & Forouzan, Data Structures: A Pseudo-code approach with C, 2nd Ed, Cengage Learning,2014.
6.	Jean-Paul Tremblay & Paul G. Sorenson, An Introduction to Data Structures with Applications, 2 nd Ed, McGraw Hill, 2013
7.	A M Tenenbaum, Data Structures using C, PHI, 1989
8.	Robert Kruse, Data Structures and Program Design in C, 2nd Ed, PHI, 1996.
9.	http://opendatastructures.org, https://donsheehy.github.io/datastructures

CO-PO Mapping												
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	-	-	-	1	-	-	-	-	-	-	2
CO2	3	3	3	-	-	-	-	-	1	-	1	2
CO3	2	2	2	1	3	-	-	-	-	-	1	3
CO4	3	2	3	-	-	-	-	-	-	2	3	2
CO5	3	2	3	-	-	-	-	-	-	2	3	2

High-3, Medium-2, Low-1

Course Title	**Object Oriented Programming**	Semester	III
Course Code	**MVJ19CS33**	**CIE**	50
Total No. of Contact Hours	60 L: T : P :: 40 : 0 : 20	**SEE**	50
No. of Contact Hours/week	5	**Total**	100
Credits	3	**Exam. Duration**	3 Hours

Course objective is to:

This course will enable students to

- Identify the need for Java - an object oriented language. Set up Java JDK environment to create, debug and run simple Java programs.
- Illustrate the use of classes and distinguish the usage of different types of Inheritance and constructors in real world.
- Demonstrate the use of exceptions and to create multi-threaded programs
- Illustrate the use of Collections with elements in Java program.
- Develop Java Application using JDBC connectivity.

Module-1	**RBT Level** L1,L2,L3	8Hrs.

Prerequisites : Basic Knowledge about C or C++

Introduction to Object Oriented Concepts and Java: Java's Magic: the Byte code; Java Development Kit (JDK); The Java Buzz words, Object Oriented Programming - Two Paradigms, Abstraction, The Three OOP Principles and its advantages, Simple Java programs. Data types, variables and arrays, Operators, Control Statements.

Laboratory Sessions/ Experimental learning:

A professor in college will allow a student to be excused from the final exam if either of the following is true: • They have a 90% average or higher in the class and have missed 3 or less class lectures. • They have a 80% average or higher in the class and have not missed any class lectures. The program below will determine whether a student can get out of the exam or not. Rewrite the program so only one if statement is used.

Applications:

Arrays in mathematical vectors, matrices.

Video link / Additional online information (related to module if any):

Differences between JVM vs JRE vs JDK in Java:
https://www.youtube.com/watch?v=5Bp6GLU6HKE

Module-2	**RBT Level** L2,L3	8Hrs.

Classes, Inheritance, Packages and Interfaces: Classes fundamentals; Declaring objects; Assigning object reference variables; Introducing Methods, Constructors, this keyword, Finalize Method. Inheritance: Inheritance basics, using super, creating multi-level hierarchy ,when constructors are called, method overriding, using abstract classes. Packages, Access Protection, Importing Packages, Interfaces.

Laboratory Sessions/ Experimental learning:

Write a program that calculates the number of buckets of paint to use for a room and the optimal number of cans to purchase. You need to ask the height of the room and the length and width of the room. The room is rectangular. You must paint the walls and the ceiling but not the floor. There are no windows or skylights. You can purchase the following size buckets of paint.

- 5-liter bucket costs $15 each and covers 1500 square feet.
- 1-liter bucket costs $4 and covers 300 square feet.

Applications:

Inheritance in Banking Sectors

Video link / Additional online information (related to module if any):

Types of Inheritance:

https://www.youtube.com/watch?v=ZP27c7i5zpg

Module-3	**RBT Level** L2,L3,L4	8Hrs.

Exception Handling and Multi-Threaded Programming : Exception Handling fundamentals, Exception Types, Uncaught Exceptions, Using try catch, Multiple catch clauses, Nested try statements, throw, throws, finally, Java's built-in exceptions, Programming Examples.

Multi-Threaded Programming: The java thread model, Main thread, Creating Thread, Creating multiple threads, Using isAlive() and join(),Thread priorities, Synchronization; InterThread Communication - Bounded buffer problem.

Laboratory Sessions/ Experimental learning:

The Producer-Consumer problem describes two processes, the producer and the consumer, which share a common, fixed-size buffer used as a queue.

> The producer's job is to generate data, put it into the buffer, and start again.
> At the same time, the consumer is consuming the data (i.e. removing it from the buffer), one piece at a time.

Make sure that the producer won't try to add data into the buffer if it's full and that the consumer won't try to remove data from an empty buffer. Write a java code to get the solution for this multi-process synchronization problem.

Applications:

Multithreads in Browsers, Servers

Video link / Additional online information (related to module if any):

Multithreading:

https://www.youtube.com/watch?v=O_Ojfq-OIpM

Module-4	**RBT Level** L3,L4,L6	8Hrs.

The collections and Framework: Collections Overview, Recent Changes to Collections, The Collection Interfaces, The Collection Classes, Accessing a collection Via an Iterator, Storing User Defined Classes in Collections. **Java Lambda expressions:** Java Lambda expressions, Using Java Lambda expressions, Lambda expression vs method in java, Lambda expression in the array list.

Laboratory Sessions/ Experimental learning:

Write a Java program to iterate through all elements in a array list **.**

Write a Java program to create a new array list, add some colors (string) and print out the collection

Applications:

Elements in group

Video link / Additional online information (related to module if any):

https://www.youtube.com/watch?v=Q_9vV3H-dt4

Module-5	**RBT Level** L4,L5,L6	8Hrs.

JDBC: The Concept of JDBC; JDBC Driver Types; JDBC Packages; A Brief Overview of the JDBC process; Database Connection; Associating the JDBC/ODBC Bridge with the Database; Statement Objects; ResultSet; Transaction Processing; Metadata, Data types; Exceptions.

Laboratory Sessions/ Experimental learning:

Develop Student Management System application with swings as the front end and database as the back end using JDBC connectivity.

Applications:

Scientific Applications, Financial Applications

Video link / Additional online information (related to module if any):

Java JDBC :

https://www.youtube.com/watch?v=hEWBIJxrLBQ

Course outcomes:

After studying this course students will be able to do

C303.1	Illustrate the Object Oriented Programming concepts and basic characteristics of Java
C303.2	Demonstrate the principles of classes, inheritance, packages and interfaces
C303.3	Experiment with exception handling Mechanisms and Create multi-threaded programs
C303.4	Interpret the need for advanced Java concepts like collections in developing modular and efficient programs
C303.5	Develop an application with Database using JDBC connectivity.

Reference Books:

1.	Herbert Schildt, Java The Complete Reference, 7 /9th Edition, Tata McGraw Hill, 2007.
2.	Jim Keogh: J2EE-The Complete Reference, McGraw Hill, 2007.
3.	Effective Java, Third Edition, Joshua Bloch, Addison-Wesley Professional,2017
4.	Richard Warburton, Java 8 Lambdas: Pragmatic Functional Programming Kindle Edition.
5.	Mahesh Bhave and Sunil Patekar, "Programming with Java", First Edition, Pearson Education,2008, ISBN:9788131720806
6.	Rajkumar Buyya , S Thamarasi selvi, xingchen chu, Object oriented Programming with java, Tata McGraw Hill education private limited.
7.	E Balagurusamy, Programming with Java A primer, Tata McGraw Hill companies

CO-PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	1	-	-	-	-	-	-	-	-	3
CO2	3	3	1	-	-	-	-	-	-	-	-	3
CO3	3	3	1	2	-	-	-	-	-	1	-	3
CO4	3	3	3	3	-	-	-	2	2	2	-	3
CO5	3	3	3	3	-	-	2	2	3	2	-	3

High-3, Medium-2, Low-1

Course Title	**Unix Shell Programming**	Semester	III
Course Code	**MVJ19CS34**	**CIE**	50
Total No. of Contact Hours	60 L : T : P :: 40 : 0 : 20	**SEE**	50

No. of Contact Hours/week	4	**Total**	100
Credits	3	**Exam. Duration**	3 Hours

Course Objectives:

The Students will be able to

- To provide introduction to UNIX operating system and its File System.
- To gain an understanding of important aspects related to the shell and the process.
- To develop the ability to formulate regular expressions and use them for pattern matching.
- To provide a comprehensive introduction to Shell Programming, services and utilities.

Module-1	**RBT Level** L1, L2	10 Hrs.

Introduction to UNIX and File system:

Introduction to file system. What are the different file systems?

Why UNIX? What is CLI, how is it different than a computer program. Why do programmers around the world prefer Linux as the OS? Inside UNIX, General features of a command, PATH, Internal and External commands, Command structure. The File System: The File, What's in a (File)name, The Parent-Child relationship, The UNIX File System − hierarchical directory structure (files, inodes), pwd, Absolute pathnames, cd, Relative pathnames, mkdir, rmdir, cp, rm, mv, cat, file.

File Attributes: ls, ls −l, ls −d, File Permissions, chmod, File systems and inodes, ln: Links, Hard links, Symbolic links, find.

Practical Exposure: Commands: ^C, ^D, ^U, ^S, ^Q in a Unix terminal window

Module-2	**RBT Level** L3	10 Hrs.

Shell and Process:

What is a script? What are Shell scripts?

Free Course: https://www.udemy.com/course/linux-shell-scripting-free/

The Shell: The shell as command processor, Pattern matching, Redirection, Pipes, Command substitution, Shell variables.

The Process: Understanding the Process, How a process is created, The Login shell, init, Internal and External commands, ps, Running jobs in background, Signals, kill, Job control, cron.

Make: Handling multisource C applications, A multisource application, make.

Flipped Classroom Session: Escaping and quoting, Special Files, tee, nice, at and batch.

Module-3	**RBT Level** L3	10 Hrs.

Advanced Filters:

pr: paginating files, head: displaying the beginning of the file, tail: displaying the end of the file, cut: slitting a file vertically, paste: pasting files, sort: ordering a file, uniq: locate repeated and non-repeated lines, tr: translating characters, displaying a word count list.

Filters using Regular Expressions: grep and sed. Programming with awk: awk preliminaries, Splitting a line into fields, BEGIN and END sections, Built-in variables, Arrays, Functions, Control flow, Looping.

Module-4	**RBT Level** L2,L3	10 Hrs.

Shell Programming:

Shell Scripts, read: Making scripts interactive, Positional parameters, Exit status of command, Logical operators && and || - Conditional execution, exit status of a command, if conditional, using test to evaluate expression, case conditional, expr: Computation and String handling, Looping: using while, until and for.

Documentation by students (Assignment): man documentation, General-Purpose Utilities.

Module-5	**RBT Level** L2,L3	10 Hrs.

Services and Utilities:

The vi Editor: vi Preliminaries, Quitting vi – The Last Line Mode, Inserting and Replacing Text, Saving Text, Exit to the UNIX Shell, The Repeat Factor, The Command Mode, Navigation, Operators, Deleting, Moving and Yanking Text, Changing Text(c and -), The Dot: Repeating the Last Command, Undoing Last Editing Instructions(u and U), String Search, Searching with Regular Expressions, Search and Replace (:s). ftp: File Transfer Protocol, The Mail Service, The Web Service.

Demonstration: Web server setup and configuration; gdb.

Laboratory Sessions

1. Practical Exposure: Commands: ^C, ^D, ^U, ^S, ^Q in a Unix terminal window

2. Free Course: https://www.udemy.com/course/linux-shell-scripting-free/

Course outcomes:

CO1	Describe the architecture and features of the UNIX operating system and distinguish it from other operating systems [L1, L2].
CO2	Demonstrate UNIX commands for file handling and process control [L3].
CO3	Construct regular expressions for pattern matching and apply them to various filters for a specific task [L3].
CO4	Analyse a given problem and apply requisite facets of shell programming in order to devise

	a shell script to solve the problem [L2, L3].
CO5	Apply the services and utilities provided by the Unix Shell Programming for various software development needs [L2,L3].

Reference Books:	
1.	Sumitabha Das: "YOUR UNIX – The Ultimate Guide", Tata McGraw Hill, 23rd reprint, 2012
2.	"Sumitabha Das: "UNIX – Concepts and Applications", 4th Edition, Tata McGraw Hill, Copyright ©2006
3.	Behrouz A. Forouzan and Richard F. Gilberg: "UNIX and Shell Programming", Cengage Learning, 2005.
4.	M.G. Venkateshmurthy: "UNIX & Shell Programming", Pearson Education, 2005.

CO-PO Mapping												
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	1	-	-	-	-	-	1	-	2	-	-
CO2	2	1	2	1	-	1	-	1	-	2	-	1
CO3	2	3	3	2	-	1	-	1	-	1	-	2
CO4	2	3	2	3	-	1	1	1	-	-	1	2
CO5	2	2	3	3	-	1	2	1	2	-	1	2

High-3, Medium-2, Low-1

Course Title	**Computer Organization and Architecture**	**Semester**	III
Course Code	**MVJ19CS35/IS35**	**CIE**	50
Total No. of Contact Hours	60 L: T : P :: 40 : 0 : 20	**SEE**	50
No. of Contact Hours/week	4	**Total**	100
Credits	03	**Exam. Duration**	3.00 Hrs

Course objective is to:

- To learn the basic structure and operations of a computer. To learn the arithmetic and logic unit.
- To learn the different ways of communication with I/O devices & memories, memory hierarchies, cache memories and virtual memories.
- To understand & implement arithmetic process.
- To understand the processor and pipelining concepts.
- To understand parallelism and multi-core processors.

Module-1	**RBT Level**	10Hrs.

	L1,L2,L3	
Basic Structure of Computers: Basic Operational Concepts, Bus Structures, Performance –Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement. **Machine Instructions and Programs:** Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes, Assembly Language, Basic Input and Output Operations, Stacks and Queues, Subroutines, Additional Instructions, Encoding of Machine Instructions. **Arithmetic:** Numbers, Arithmetic Operations and Characters, Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed Operand Multiplication, Fast Multiplication, Integer Division. **Text book 1: Chapter 1 – 1.1 to 1.9,Chapter2 – 2.1 to 2.10** **Text book 1: Chapter6 – 6.1 to 6.7** **Laboratory Sessions/ Experimental learning:** Study of peripherals, components of a Computer System **Applications:** Basic Computer Devices **Video link :** https://nptel.ac.in/courses/106105163/		
Module-2	**RBT Level** L1,L2,L3	10Hrs.
Input/output Organization: Accessing I/O Devices, Interrupts – Interrupt Hardware, Direct Memory Access, Buses, Interface Circuits. Standard I/O Interfaces – PCI Bus, SCSI Bus, USB **Text book 1: Chapter4 – 4.1 to 4.7** **Laboratory Sessions/ Experimental learning:** Design of ALU **Applications:** input /output operations Videolink:https://www.youtube.com/watch?v=RkAE4zE4uSE&list=PL13FD5F00C21BBC0B&index=11		
Module-3	**RBT Level** L1,L2,L3	10Hrs.
Memory: Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories – Types of cache ,Cache miss management Mapping Functions, Replacement Algorithms, Performance Considerations,(ARM Cache and Pentium cache). **Text book 1: Chapter5 – 5.1 to 5.4, 5.5** **Laboratory Sessions/ Experimental learning:** Design of Memory **Applications:** Different Types of Memory Video link : https://nptel.ac.in/courses/106105163/		
Module-4	**RBT Level** L1,L2,L3	10Hrs.
Processor : A Basic MIPS implementation – Building a Data path – Control Implementation Scheme – Pipelining – Pipelined data path and control – Handling Data Hazards & Control Hazards –Exceptions. **Text book 2: Chapter 4**. **Laboratory Sessions:** Instruction scheduling **Applications:** Types of processor **Video link:** https://nptel.ac.in/courses/106106166/		
Module-5	**RBT Level** L1,L2,L3	10Hrs.
Parallelism: Parallel processing challenges –Flynn‘s classification – SISD, MIMD, SIMD, SPMD, and Vector Architectures - Hardware multithreading – Multi-core processors and other Shared Memory Multiprocessors - Introduction to Graphics Processing Units, Clusters, Warehouse Scale Computers and		

other Message-Passing Multiprocessors.

Text book 2: Chapter 6.

Laboratory Sessions : Process Scheduling

Applications: Grid and Cloud Computing

Video link: https://nptel.ac.in/courses/106102114/

Course outcomes:	
C203.1	Explain the basic organization of a computer system.
C203.2	Demonstrate functioning of different sub systems, such as processor, Input/output, and memory.
C203.3	Design and analyses simple arithmetic and logical units.
C203.4	Illustrate hardwired control and micro programmed control, pipelining, embedded and other Computing systems.
C203.5	Design and analyses of simple Parallelism and Multithread.

Reference Books:	
1.	Carl Hamacher, Zvonko Vranesic, SafwatZaky, Computer Organization, 5th Edition, Tata McGraw Hill, 2002. (Listed topics only from Chapters 1, 2, 4, 5, and 6).
2.	David A. Patterson and John L. Hennessy, Computer Organization and Design: The Hardware/Software Interface, Fifth Edition, Morgan Kaufmann / Elsevier, 2014.(Listed topics only from Chapters 4and 6).
3	John P. Hayes, Computer Architecture and Organization, Third Edition, Tata McGraw Hill, 2012.
4	John L. Hennessey and David A. Patterson, Computer Architecture – A Quantitative Approach‖, Morgan Kaufmann / Elsevier Publishers, Fifth Edition, 2012.
5	http://vlabs.iitkgp.ac.in/coa/

CO-PO Mapping												
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
C203.1	1	2	1	1	1	-	-	-	-	-	-	-
C203.2	2	2	1	1	1	-	-	-	-	-	--	-
C203.3	1	2	2	1	1	-	-	-	-	-	-	-
C203.4	2	2	2	1	2	-	-	-	-	-	-	-
C203.5	1	2	2	1	2	-	-	-	-	-	-	-

High-3, Medium-2, Low-1

Course Title	**Analog and Digital Electronics**	Semester	III
Course Code	**MVJ19CS36/IS36**	**CIE**	50
Total No. of Contact Hours	60 L : T : P :: 40 : 0 : 20	**SEE**	50
No. of Contact Hours/week	6	**Total**	100
Credits	3	**Exam. Duration**	3 Hours

Course objective is to:

The students will be able to

- Analyse the working of oscillators and use of regulators.
- Make use of simplifying techniques in the design of combinational circuits.
- Illustrate combinational and sequential digital circuits.
- Demonstrate the use of flipflops and design registers and counters.
- Design and test Analog-to-Digital and Digital-to-Analog conversion techniques.

Module-1	**RBT Level** L2	10 Hrs.

Prerequisites : Basic analog Circuits

Metal Oxide Semiconductor Field Effect transistor(MOSFET): Structure and I-V characteristics, MOSFET as a switch, MOSFET as an amplifier, CMOS and its applications.

Oscillators: Basic working and applications of RC Phase shift oscillator, Wien Bridge oscillator, LC oscillator, Colpitt oscillator, Crystal Oscillator.

Linear Power Supplies: Constituents of a Linear Power Supply, Designing Mains Transformer, Linear IC voltage regulators, Regulated Power Supply Parameters

Module-2	**RBT Level** L2,L3	10 Hrs.

Prerequisites:Digital Electronic Fundamentals

Karnaugh maps:Minimum forms of switching functions, two and three variable Karnaugh maps, four variable karnaugh maps, Quine-McClusky Method: determination of prime implicants, The prime implicant chart, petricks method, simplification of incompletely specified functions, simplification using map-entered variables

Activity: Writing and Analyzing C program for K-maps.

Module-3	**RBT Level** L2,L3	10Hrs.

Combinational Circuits: Multiplexer, Decoders, Adders, Subtractors, BCD arithmetic, carry look ahead adder, serial adder, ALU-Design and popular MSI chips, digital comparator, parity

checker/generator, code converters, priority encoders, decoders/drivers for display devices,

Activity: Designing a 32-bit ALU

Module-4	**RBT Level** L2,L3	10 Hrs.

Flip-Flops and Registers:

Flip Flops: S-R,J-K,D and T flip flops,Edge-triggered JK FLIP-FLOPs

Registers: Types of Registers, Serial In - Serial Out, Serial In - Parallel out, Parallel In - Serial Out, Parallel In - Parallel Out, Universal Shift Register, Applications of Shift Registers.

Counters: Asynchronous Counters, Decoding Gates, Synchronous Counters, Changing the Counter Modulus, Decade Counters, Applications of Counters.

Activity: Implementing 2 digit counters using seven segment display

Module-5	**RBT Level** L2	10 Hrs.

D/A Conversion and A/D Conversion:

Digital to analog converters: weighted resistor/converter, R-2R Ladder D/A converter, specifications for D/A converters, examples of D/A converter lCs, sample and hold circuit.

Analog to digital converters: quantization and encoding, parallel comparator A/D converter, successive approximation A/D converter, counting A/D converter, dual slope A/D converter, A/D converter using voltage to frequency and voltage to time conversion, specifications of A/D converters, example of A/D Converter ICs

Activity: Demonstration of CODEC which houses both ADC and DAC.

Laboratory Sessions

Plotting the V-I characteristics of MOSFET

Implementing adders and subtarctors

Implementing the simplified equation obtained from K-maps and verify with the truth table

Course outcomes:

CO1	Design and analyze analog circuits using transistors,power supply, MOSFETS,regulator IC and opamp.
CO2	Simplify digital circuits using Karnaugh Map , POS and Quine-McClusky Methods
CO3	Explain construction and working of data processing circuits
CO4	Understanding the various types of latches and flip flops and building the registers and counters using flip flops.
CO5	Explain the basic principles of A/D and D/A conversion circuits and develop the same.

Reference Books:	
1.	Anil K Maini, Varsha Agarwal, Electronic Devices and Circuits, Wiley, 2012.
2.	Charles H Roth and Larry L Kinney, Fundamentals of Logic design, Cengage Learning,2019.
3.	Donald P Leach, Albert Paul Malvino & Goutam Saha, Digital Principles and Applications, 8th Edition, Tata McGraw Hill, 2015.
4.	M. Morris Mani, Digital Design, 4th Edition, Pearson Prentice Hall, 2008.
5.	David A. Bell, Electronic Devices and Circuits, 5th Edition, Oxford University Press, 2008

CO-PO Mapping												
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	-	-	-	-	-	-	-	1
CO2	3	3	2	2	-	-	-	-	-	-	-	1
CO3	3	3	3	2	-	-	-	-	-	-	-	1
CO4	3	3	2	2	-	-	-	-	-	-	-	1
CO5	3	3	3	2	-	-	-	-	-	-	-	1

High-3, Medium-2, Low-1

Course Title	**Data Structures and Applications Laboratory**	Semester	III
Course Code	MVJ19CSL37	**CIE**	50
Total No. of Contact Hours	40 L : T : P :: 10 : 0 : 30	**SEE**	50
No. of Contact Hours/week	3	**Total**	100
Credits	2	**Exam. Duration**	3 Hours

Course objective:

This course will enable students to get practical experience in design, develop, implement, analyze and evaluation of

- Linear data structures and their applications such as stacks, queues and lists,
- Non-Linear data structures and their applications such as Trees &Graphs
- Sorting and Hashing techniques

Sl.No	**Experiment**	**RBT Level**

<table>
<tr>
<td>1</td>
<td>A courier company has number of items to be delivered to its intended customers through its salesman. The salesman visits the following cities to deliver the respective items. Write a C program,
<table>
<tr><th>S.No</th><th>Cities</th><th>Number of items</th></tr>
<tr><td>1</td><td>Agra</td><td>25</td></tr>
<tr><td>2</td><td>Chennai</td><td>50</td></tr>
<tr><td>3</td><td>Kolkata</td><td>59</td></tr>
<tr><td>4</td><td>Mumbai</td><td>72</td></tr>
<tr><td>5</td><td>Delhi</td><td>12</td></tr>
</table>
a) To display name of cities where salesman has delivered maximum and minimum number of items
b) To search the number of items to be delivered of a user supplied city.</td>
<td>L3</td>
</tr>
<tr>
<td>2</td>
<td>Implement Knuth-Morris- Pratt pattern matching algorithm using C program.</td>
<td>L3</td>
</tr>
<tr>
<td>3</td>
<td>Design, Develop and Implement a menu driven Program in C with the listed operations for the data structure which follows Last In First Out (LIFO) order. (Use Array Implementation of specified DS with maximum size MAX).
a. Push an Element
b. Pop an Element
c. Demonstrate how it can be used to check Palindrome
d. Demonstrate Overflow and Underflow situations
e. Display the status
f. Exit
Support the program with appropriate functions for each of the above operations</td>
<td>L3</td>
</tr>
<tr>
<td>4</td>
<td>Design, Develop and Implement a Program in C for converting an Infix Expression to Postfix Expression. Program should support for both parenthesized and free parenthesized expressions with the operators: +, -, *, /, % (Remainder), ^ (Power) and alphanumeric operands.</td>
<td>L3</td>
</tr>
<tr>
<td>5</td>
<td>Design, Develop and Implement a menu driven Program in C for the following operations on Ring Buffer of Integers (Use Array Implementation)
a. Insert an Element on to Ring Buffer</td>
<td>L3</td>
</tr>
</table>

	b. Delete an Element from Ring Buffer c. Demonstrate Overflow and Underflow situations on Ring Buffer d. Display the status of Ring Buffer e. Exit Support the program with appropriate functions for each of the above operations	
6	Design, Develop and Implement a menu driven Program in C for the following operations on Singly Linked List (SLL) of Student Data with the fields: USN, Name, Programme, Sem, PhNo a. Create a SLL of N Students Data by using front insertion b. Display the status of SLL and count the number of nodes in it c. Perform Insertion / Deletion at End of SLL d. Perform Insertion / Deletion at Front of SLL e. Exit	**L3**
7	Design, Develop and Implement a menu driven Program in C for the following operations on Doubly Linked List (DLL) of Employee Data with the fields: SSN, Name, Dept, Designation, Sal, PhNo. a. Create a DLL of N Employees Data by using end insertion. b. Display the status of DLL and count the number of nodes in it. c. Perform Insertion and Deletion at End of DLL . d. Perform Insertion and Deletion at Front of DLL . e. Demonstrate how this DLL can be used as Double Ended Queue. f. Exit	**L3**
8	Design, Develop and Implement a menu driven C Program for the following operations on Binary Search Tree (BST) of Integers. a) Create a BST of N Integers: 6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2. b) Traverse the BST recursively in inorder, preorder & postorder c) Search the BST for a given element (KEY) and report the appropriate message	**L3**

9	Design, Develop and Implement a Program in C for the following operations on Graph(G) of Cities a. Create a Graph of N cities using Adjacency Matrix. b. Print all the nodes reachable from a given starting node in a digraph using DFS/BFS method	**L3**
10	Develop a C program to sort a given set of n integer elements using Quick Sort method. Run the program for varied values of n and show the results of each iteration.	**L3**
11	Given a File of N employee records with a set K of Keys(4-digit) which uniquely determine the records in file F. Assume that file F is maintained in memory by a Hash Table(HT) of m memory locations with L as the set of memory addresses (2-digit) of locations in HT. Let the keys in K and addresses in L are Integers. Design and develop a Program in C that uses Hash function H: K →L as H(K)=K mod m (remainder method), and implement hashing technique to map a given key K to the address space L. Resolve the collision (if any) using linear probing.	**L3**

Course outcomes:

CO1	Analyze and Compare various linear data structures
CO2	Code, debug and demonstrate the working nature of different types of data structures and t applications
CO3	Implement, analyse and evaluate the searching and sorting algorithms
CO4	Choose the appropriate data structure for solving real world problems

---

CO-PO Mapping												
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	-	3	3	-	-	3	-	3	2
CO2	3	3	2	-	3	3	-	-	3	-	3	2
CO3	3	3	2	-	3	3	-	-	3	-	3	2
CO4	3	3	2	-	3	3	-	-	3	-	3	2

High-3, Medium-2, Low-1

Course Title	**Analog and Digital Electronics Lab**	Semester	III
Course Code	**MVJ19CSL38**	**CIE**	50
Total No. of Contact Hours	40 L : T : P :: 10 : 10 : 20	**SEE**	50
No. of Contact Hours/week	3	**Total**	100
Credits	2	**Exam. Duration**	3 Hours

Course objective is to:

The Students will be able to

- Analog components and circuits including transistor,regulator, etc.
- Combinational logic circuits.
- Flip - Flops and their operations
- Counters and Registers using Flip-flops.
  Synchronous and Asynchronous Sequential Circuits

Laboratory Sessions

Sl. No	**Experiment**	**RBT Level**
1	Study of transistor phase shift oscillator and observe the effect of variation in R & C on oscillator frequency and compare with theoretical value.	**L2**
2	Design and test IC 723 voltage regulator	**L3**
3	Given a 4-variable logic expression, simplify it using Entered Variable Map and realize the simplified logic expression using 8:1 multiplexer IC.	**L2**
4	Design and implement a faster way3 to add binary numbers using carry look ahead adders.	**L3**
5	a) Realization and implementation of 2-bit comparator using logic gates. b) Implementation of 4-bit magnitude comparator using IC 7485.	**L3**
6	To design and construct basic flip-flops R-S ,J-K,J-K Master slave flip-flops using gates and verify their truth table	**L3**
7	Implementation of SISO, SIPO, PISO and PIPO shift registers using Flip-flops	**L3**
8	Design and implementation of 3-bit synchronous up/down counter	**L3**
9	Design and implement a ring counter and Johnson counter using 4-bit shift register and demonstrate its working.	**L3**
10	Design and implement a mod-n (n<8) synchronous up counter using J-K Flip-Flop ICs and demonstrate its working.	**L3**
11	Design and implement an asynchronous counter using decade counter IC to count up from 0 to n (n<=9) and demonstrate on 7-segment display	**L3**

	(using IC-7447).	
12	Design 4 bit r-2r ladder DAC using opamp.	**L3**

Course outcomes:	
CO1	Demonstrate various Electronic Devices like Cathode ray Oscilloscope, Signal generators, Digital Trainer Kit, Multimeters and components like Resistors, Capacitors, Op amp and Integrated Circuit
CO2	Examine and verify different analog circuits.
CO3	Design and demonstrate various combinational logic circuits.
CO4	Design and demonstrate various types of counters and Registers using Flip-flops

---

CO-PO Mapping												
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	-	-	-	-	-	-	-	-
CO2	3	3	2	2	-	-	-	-	-	-	-	-
CO3	3	3	3	2	-	-	-	-	-	-	-	-
CO4	3	3	2	2	-	-	-	-	-	-	-	-

High-3, Medium-2, Low-1

Course Title	**Operations Research, Numerical and Statistical Methods**	Semester	IV
Course Code	MVJ19MCS41/MIS41	**CIE**	50
Total No. of Contact Hours	60 L : T : P :: 40 : 0 : 20	**SEE**	50
No. of Contact Hours/week	4	**Total**	100
Credits	3	**Exam. Duration**	3 HOURS

Course objective is to:

The purpose of this course is to make students well conversant with numerical methods to solve ordinary differential equations, complex analysis, sampling theory Operational research emerging in science and engineering.

Module-1	**RBT Level** L1,L2,L3	10 Hrs.

NUMERICAL METHODS-1

Numerical solution of Ordinary Differential Equations of first order and first degree: Picard's

method, Taylor's series method, Modified Euler's method, Runge-Kutta method of fourth order, Predictor and Corrector method: Milne's Method.		
Module-2	**RBT Level** L1,L2,L3	10 Hrs.
NUMERICAL METHODS-2: Numerical solution of Ordinary Differential Equations of second order : Picard's method, Runge-Kutta method of fourth order, Predictor and Corrector method: Milne's Method. **Calculus of Variations:** Variation of function and Functional, variational problems. Euler's equation, Geodesics, hanging chain, problems.		
Module-3	**RBT Level** L1,L2,L3	10 Hrs.
OPERATIONS RESEARCH-1 Introduction to Linear Programming Problem (LPP): Prototype example, Assumptions of LPP, Formulation of LPP and Graphical method various examples. the simplex method, Big M method, Two phase method and dual simplex method		
Module-4	**RBT Level** L1,L2,L3	10 Hrs.
OPERATIONS RESEARCH-2 The transportation problem: Initial Basic Feasible Solution (IBFS) by North West Corner Rule method, Matrix Minima Method, Vogel's Approximation Method. **Game Theory**: The formulation of two persons, zero sum games; saddle point, maximin and minimax principle, Solving simple games- a prototype example; Games with mixed strategies; Graphical solution procedure		
Module-5	**RBT Level** L1,L2,L3	10 Hrs.
STATISTICAL METHODS Fitting of the curves of by the method of least square, Correlation and Regression , Regression coefficients, line of regression problems. Curve fitting by the method of least squares, Fitting of the curves of the form = + , = 2 + + , = .		
Course outcomes:		
CO1	Solve first and second order ordinary differential equation arising in flow problems using single step and multistep numerical methods.	
CO2	Determine the extremals of functionals and solve the simple problems of the	

	calculus of variations.
CO3	Solve the mathematical formulation of linear programming problem.
CO4	Solve the applications of transport problems and theory of games.
CO5	Fit a suitable curve by the method of least squares and determine the lines of regression for a set of statistical data.

Reference Books:	
1.	B.S. Grewal, "Higher Engineering Mathematics" Khanna Publishers, 43rd Edition, 2013.
2.	Erwin Kreyszig, "Advanced Engineering Mathematics", Wiley-India publishers, 10th edition,2014.
3	Ramana B. V., "Higher Engineering Mathematics", Tata Mc Graw-Hill, 2006.
4	Bali N. P. & Manish Goyal, "A text book of Engineering Mathematics", Laxmi Publications, 8th Edition
5	Jain R. K. & Iyengar S.R.K., Advanced Engineering Mathematics, Narosa Publishing House, 2002.
6	S. D. Sharma, "Operations Research", Kedar Nath and Ram Nath Publishers, Seventh Revised Edition 2014.

CO-PO Mapping												
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	0	3	0	0	0	0	0	0	0	1
CO2	3	2	0	3	0	0	0	0	0	0	0	0
CO3	3	3	0	2	0	0	0	0	0	0	0	0
CO4	2	3	0	3	0	0	0	0	0	0	0	1
CO5	3	3	0	3	0	0	0	0	0	0	0	1

High-3, Medium-2, Low-1

Course Title	Analysis and Design of Algorithms	Semester	IV
Course Code	**MVJ19CS42/IS42**	**CIE**	50
Total No. of Contact Hours	60 L : T : P :: 40 : 0 : 20	**SEE**	50
No. of Contact Hours/week	4	**Total**	100
Credits	03	**Exam. Duration**	03 hours

Course objective is to:

- Identify the importance of different asymptotic notation.
- Determine the complexity of recursive and non-recursive algorithms.
- Compare the efficiency of various design techniques like greedy method, backtracking etc.
- Apply appropriate method to solve a given problem.

Module-1	**RBT Level** L1, L2	11 Hrs.

Basic Concept of Algorithms: Introduction-What is an Algorithm, Algorithm Specification, Analysis Framework, Performance Analysis: Space complexity, Time complexity. Asymptotic Notations: Big-Oh notation (O), Omega notation (Ω), Theta notation (Θ), and Little-oh notation (o), Mathematical analysis of Non-Recursive and recursive Algorithms with Examples . Important Problem Types. Fundamental Data Structures

Applications: developing computational tools and bioinformatics software, Mathematics

Video link / Additional online information (related to module if any):

1. http://www.nptelvideos.com/video.php?id=1442
2. https://nptel.ac.in/courses/106105085/

Module-2	**RBT Level** L2, L4	10 Hrs.

Simple Design Techniques – Brute force : Selection sort, Bubble sort,Sequential Search and Brute-Force String Matching , Exhaustive search –Traveling Salesman problem, Knapsack problem , Assignment Problem. **Divide and Conquer:** General method, Binary search, Recurrence equation for divide and conquer, Finding the maximum and minimum , Merge sort, Quick sort , Strassen's matrix multiplication , Advantages and Disadvantages of divide and conquer.

Applications: power distribution (electrical field), Online shopping and delivery (real time)

Video link / Additional online information (related to module if any):

1. https://nptel.ac.in/courses/106102064/
2. https://www.youtube.com/watch?v=MFfD57DTDQY

Module-3	**RBT Level** L3	13 Hrs.

Decrease and Conquer approach: Topological Sort, Decrease-by-a-Constant-Factor Algorithms: Josephus Problem

Greedy Method: General method, Coin Change Problem, Knapsack Problem, Job sequencing with deadlines. Minimum cost spanning trees: Prim's Algorithm, Kruskal's Algorithm. Single source

shortest paths: Dijkstra's Algorithm. Huffman Trees and Codes. Laboratory Sessions/ Experimental learning: Solving real time problems using Greedy Technique. Applications: Optimization Problems Video link : https://nptel.ac.in/courses/106/106/106106131/

Module-4	**RBT Level** L2	13 Hrs.
Dynamic Programming: General method with Examples, Multistage Graphs. Transitive Closure: Warshall's Algorithm, All Pairs Shortest Paths: Floyd's Algorithm, Optimal Binary Search Trees, Knapsack problem, Bellman-Ford Algorithm , Travelling Sales Person problem , Reliability design. Laboratory Sessions/ Experimental learning: Solving real time problems using Dynamic Programming. Applications: Computer Networks. Video link: https://nptel.ac.in/courses/106/106/106106131/		

Module-5	**RBT Level** L1, L3	13 Hrs.
Backtracking: General method, N-Queens problem, Sum of subsets problem, Graph coloring, Hamiltonian cycles Programme and Bound: Assignment Problem, Travelling Sales Person problem, 0/1 Knapsack problem : LC Programme and Bound solution : FIFO Programme and Bound solution. NP-Complete and NP-Hard problems: Basic concepts, non-deterministic algorithms, P, NP, NP-Complete, and NP-Hard classes. Laboratory Sessions/ Experimental learning: Solving real time problems using Backtracking Technique. Applications: To solve puzzles such as crosswords, Sudoku etc. Video link: https://nptel.ac.in/courses/106/106/106106131/		

Course outcomes:

CO1	Describe the need of algorithm and the notations used in design analysis.
CO2	Compare the efficiency of brute force, divide and conquer techniques for problem solving.
CO3	Ability to apply greedy algorithms, hashing and string matching algorithms
CO4	Ability to design efficient algorithms using various design techniques
CO5	Ability to apply the knowledge of complexity classes P, NP, and NP Complete and prove certain problems are NP-Complete

Reference Books:

1.	Introduction to the Design and Analysis of Algorithms, Anany Levitin:, 2rd Edition, 2009. Pearson.
2.	Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 3rd Edition, PHI.

3.	Design and Analysis of Algorithms, S. Sridhar, Oxford (Higher Education).
4.	http://jeffe.cs.illinois.edu/teaching/algorithms/
5.	Computer Algorithms/C++, Ellis Horowitz, Satraj Sahni and Rajasekaran, 2nd Edition, 2014, Universities Press

CO-PO Mapping												
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	--	--	--	1	--	--	--	--	--	--	2
CO2	2	3	--	--	2	--	--	--	--	--	--	--
CO3	3	3	2	2	--	--	--	--	--	--	--	2
CO4	3	3	3	--	--	--	--	--	1	--	--	2
CO5	2	2	2	1	3	--	--	--	--	--	--	3

High-3, Medium-2, Low-1

Course Title	**Software Engineering**	Semester	IV
Course Code	**MVJ19CS43**	**CIE**	50
Total No. of Contact Hours	60 L : T : P :: 40 : 0 : 20	**SEE**	50
No. of Contact Hours/week	5	**Total**	100
Credits	3	**Exam. Duration**	3 HOURS

Course objective is to:

The students will be able to

- To understand principles, concepts, methods, and techniques of the software engineering approach to producing quality software (particularly for large, complex systems).
- To impart skills in the design and implementation of efficient software systems across disciplines.
- To familiarize engineering practices and standards used in developing software products and components.
- To gather knowledge on various software testing, maintenance methods.

Module-1	**RBT Level** L1,L2,L3	10 Hrs.

FUNDAMENTALS OF SOFTWARE ENGINEERING AND REQUIREMENTS ENGINEERING

Software Engineering Fundamentals; Software processes: Software life-cycle models; Software requirements and specifications: Requirements elicitation; Requirements analysis modeling

techniques; Functional and non-functional requirements; User requirements, System requirements, requirement validation and software requirement specification document. Prototyping - Basic concepts of formal specification techniques.

Laboratory Sessions/ Experimental learning:
To write the SRS for the given real time application using report writing tools.
Applications: In Software development process.
Video link / Additional online information: https://nptel.ac.in/courses/106105182/

Module-2	**RBT Level** L1,L2,L3	10 Hrs.

SOFTWARE DESIGN

Fundamental design concepts and principles; Design characteristics; System Models - Context, Behavioral, Data and, Object models, Architectural design- System structuring, Control models; Structured design; Object-oriented analysis and design; User interface design; Design for reuse; Design patterns;

Laboratory Sessions/ Experimental learning:

Draw a class diagram, object diagram, Use case diagram, Sequence diagram and activity diagram for the given real time application using rational rose tool.
Applications: In Software development process.
Video link / Additional online information:
https://www.coursera.org/lecture/client-needs-and-software-requirements/3-2-4-use-cases-bZNCr

Module-3	**RBT Level** L1,L2,L3	10 Hrs.

SOFTWARE VALIDATION AND MAINTENANCE

Software validation: Validation planning; Testing fundamentals, including test plan creation and test case generation; Black-box and white-box testing techniques; Unit, integration, validation, and system testing; Object-oriented testing; Inspections. **Software evolution:** Software maintenance; Characteristics of maintainable software; Reengineering; Legacy systems; Software reuse.

Laboratory Sessions/ Experimental learning:
Using Selenium IDE write a test suite containing minimum 4 test cases.
Applications: In Software development process.
Video link / Additional online information: https://www.youtube.com/watch?v=T3q6QcCQZQg

Module-4	**RBT Level** L1,L2,L3	10 Hrs.

COMPONENT BASED SOFTWARE ENGINEERING

Engineering of Component-Based Systems; The CBSE Process; Domain Engineering; Component-

Based Development; Classifying and Retrieving Components; Economics of CBSE

Laboratory Sessions/ Experimental learning: Create a project using MS projects for any real time scenario.
Applications: In Software development process.
Video link / Additional online information: https://youtu.be/tIZ1dg4pxCE

Module-5	**RBT Level** L1,L2,L3	10 Hrs.

SOFTWARE QUALITY PROCESS IMPROVEMENT Overview of Quality management and Process Improvement; Overview of SEI -CMM, ISO 9000, CMMI, PCMM, TQM and Six Sigma; overview of CASE tools. Software tools and environments: Programming environments; Project management tools; Requirements analysis and design modelling tools; testing tools; Configuration management tools;

Laboratory Sessions/ Experimental learning:
Estimation of test coverage metrics using manual test metrics.
Applications: In Software development process.
Video link / Additional online information: https://nptel.ac.in/courses/110105039/

Course outcomes:

CO1	Comprehend software development life cycle and Prepare SRS document for a project
CO2	Apply software design and development techniques
CO3	Identify verification and validation methods in a software engineering project
CO4	Apply on Component based software development process.
CO5	Involve in continuous learning to solve issues of process and software product using the advanced CASE tools and techniques.

Reference Books:

1.	Ian Sommerville, "Software Engineering", 9th Edition, Addison- Wesley, 2011
2.	R. S. Pressman, Software Engineering, a practitioner's approach, McGraw Hill,7th Edition, 2010
3.	Rajib Mall, “Fundamentals of Software Engineering”, PHI Publication, 3rd edition, 2009
4.	Pankaj Jalote: An Integrated Approach to Software Engineering, Wiley India

CO-PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	2	2	2	-	-	1	2	2	2	-
CO2	2	2	2	2	2	1	-	1	2	2	2	1
CO3	2	2	2	2	2	1	-	1	2	2	2	-

CO4	1	2	2	2	2	1	-	1	2	2	2	1
CO5	1	2	2	1	2	1	2	1	2	2	2	2

High-3, Medium-2, Low-1

Subjects Interdisciplinary connectivity:

The software engineering is used for all the software development process irrespective of field and branch. Wherever the software should be created, there is a need of software engineering concepts. So it's not only related to the development process of only computer science related software and products.

Course Title	**Operating Systems**	Semester	IV
Course Code	**MVJ19CS44**	**CIE**	50
Total No. of Contact Hours	60 L : T : P :: 40 : 0 : 20	**SEE**	50
No. of Contact Hours/week	4	**Total**	100
Credits	3	**Exam. Duration**	3 HOURS

Course objective is to:

The students will be able to

- Introduce concepts and terminology used in OS
- Explain threading and multithreaded systems
- Illustrate process synchronization and concept of Deadlock
- Introduce Memory and Virtual memory management, File system and storage techniques

Module-1	**RBT Level** L2	12 Hrs.

Introduction: What operating systems do; Computer System organization; Computer System architecture; Operating System operations; Distributed system; Special-purpose systems; Computing environments. Operating System Services; User - Operating System interface; System calls; Types of system calls; System programs; Operating system design and implementation; Operating System structure; Virtual machines; System boot.

Process Management: Process concept; Process scheduling; Operations on processes;Inter process communication

Module-2	**RBT Level** L2	12 Hrs.

Multi-threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues. Process Scheduling: Basic concepts; Scheduling Criteria; Scheduling Algorithms; Multiple-processor scheduling; Thread scheduling.

Process Synchronization: Synchronization: The critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization;

Monitors.

Module-3	RBT Level L3	12 Hrs.

Deadlocks : Deadlocks; System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.

Memory Management: Memory management strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation

Module-4	RBT Level L3	12 Hrs.

Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing.

File System, Implementation of File System: File system: File concept; Access methods; Directory structure; File system mounting; File sharing;

Implementing File system: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management.

Module-5	RBT Level L3	12 Hrs.

Mass Storage Structure-Disk Structure-Disk Attachment-Disk Scheduling-Disk Management-Swap-Space Management.

Protection: Domain of protection, Access matrix, Implementation of access matrix, Access control,Revocation of access rights, Capability- Based systems.

Case Studies: Windows, Unix, Linux,Android.

Laboratory Sessions

Creating processes in Unix with commands like Fork and Exec; Pipes and process communication; Performance study of various CPU scheduling algorithms; Performance study of various Disk scheduling algorithms. Analysis various memory management techniques and page replacement policies.

Course outcomes:

CO1	Illustrate the fundamental concepts of operating systems
CO2	Compare and illustrate various process scheduling algorithms.
CO3	Ability to recognize and resolve Deadlock problems ,Memory Management techniques.
CO4	Apply appropriate memory and file management schemes.
CO5	Appreciate the need of access control and protection in Operating System and illustrate various disk scheduling algorithms.

Reference Books:

1.	Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Concepts 7th edition,Wiley-India, 2006
2.	D.M Dhamdhere, Operating Systems: A Concept Based Approach 3rd Ed, McGraw- Hill, 2013.
3.	Tanenbaum, A., "Modern Operating Systems", Prentice-Hall of India. 2004
4.	P.C.P. Bhatt, An Introduction to Operating Systems: Concepts and Practice 4th Edition,

CO-PO Mapping												
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	2	-	-	-	-	-	-	-	-	-
CO2	2	2	3	-	-	-	-	-	-	-	-	-
CO3	3	2	3	-	-	-	-	-	-	-	-	-
CO4	3	2	3	-	-	-	-	-	-	-	-	-
CO5	3	2	3	-	-	-	-	-	-	-	-	-

High-3, Medium-2, Low-1

Course Title	**Micro Controller and Embeded Systems**	Semester	IV
Courde	**MVJ19CS45**	**CIE**	50
Total No. of Contact Hours	60 L : T : P :: 40 : 0 : 20	**SEE**	50
No. of Contact Hours/week	5	**Total**	100
Credits	3	**Exam. Duration**	3 HOURS

Course objective is to:

The students will be able to

- Explain the fundamentals of ARM based system, basic hardware components, selection methods and attributes of an ARM Controller.
- Program ARM controller using the various instructions.
- Explain the fundamentals of Exception, Interrupt Handling and Memory Management Unit of ARM Controller.
- Identify the Embedded System Design applications.
- Explain the real time operating system for the embedded system design.

Module-1	**RBT Level** L1,L2,L3	10 Hrs.

ARM EMBEDDED SYSTEMS:

Prerequisites: **ARM DESIGN PHILOSOPHY,ARM DATAFLOW MODEL**

Microprocessors versus Microcontrollers, ARM Embedded Systems: The RISC design

philosophy, The ARM Design Philosophy, Embedded System Hardware, Embedded System Software.

ARM Processor Fundamentals: Registers, Current Program Status Register, Pipeline, Exceptions, Interrupts, and the Vector Table , Core Extensions

Activity:1.Comparision of Microprocessor and Microcontroller hardware Model

2.Comparing the Microprocessor and Microcontroller Software Model

Module-2	**RBT Level** L1,L2,L3	10 Hrs.

ARM Instruction Set and Programming

Prerequisites: **ARM INSTRUCTION SET,ARM ASSEMBLY PROGRAMMING**

Introduction to the ARM Instruction Set : Data Processing Instructions , Programme Instructions, Software Interrupt Instructions, Program Status Register Instructions, Coprocessor Instructions, Loading Constants

ARM programming using Assembly language: Writing Assembly code, Profiling and cycle counting, instruction scheduling

Activity: 1.Writing ARM Assembly program for Embedded System Applications

Module-3	**RBT Level** L1,L2,L3	10 Hrs.

Interrupt and Memory Management Unit:

Prerequisites:**Interrupt,Exception,Memory Management unit**

Exception, Interrupt Handling : Exception handling,Interrupts,Interrupt handling Schemes

Memory Management Unit : The Memory Hierarchy and Cache Memory, Cache Architecture, Cache Policy, Moving from MPU to an MMU,How Virtual Memory Works, Details of ARM MMU

Activity:

1) Use of External interrupt0 to turn ON/OFF led connected to Pin P1.25 of ARM Processor.

2) Use of Software Interrupt SWI instruction in programming.

3) Calculating physical memory address from logical address.

Module-4	**RBT Level** L1,L2,L3	10 Hrs.

Prerequisites: Embedded systems ,Embedded Applications

Embedded System Components: Embedded Vs General computing system, History of embedded systems, Classification of Embedded systems, Major applications areas of embedded systems, purpose of embedded systems

Core of an Embedded System including all types of processor/controller, Memory, Sensors, Actuators, LED, 7 segment LED display, stepper motor, Keyboard, Push button switch, Communication Interface (on board and external types), Embedded firmware, Other system components.

Activity:
Case Study: Digital Clock, Battery operated Smartcard Reader

Module-5	**RBT Level** L1,L2,L3	10 Hrs.

Prerequisites: Real time operating system

Real Time Operating System (RTOS) based Embedded System Design:

Operating System basics, Types of operating systems, Task, process and threads (Only POSIX Threads with an example program), Thread pre-emption, Multiprocessing and Multitasking, Task Communication (without any program), Task synchronization issues – Racing and Deadlock, Concept of Binary and counting semaphores (Mutex example without any program), How to choose an RTOS

Activity:

Case Study: Automated Meter Reading System (AMR) and Digital Camera,Real time concepts

Course outcomes:

CO1	Describe the architectural features and instructions of ARM microcontroller
CO2	Develop Assembly Programs in ARM for Embedded applications.
CO3	Describe the fundamentals of Exception, Interrupt Handling and Memory Management Unit of ARM Controller
CO4	Interface external devices and I/O with ARM microcontroller.
CO5	Demonstrate the need of real time operating system for embedded system applications

Reference Books:

1.	Andrew N Sloss, Dominic Symes and Chris Wright, ARM system developer's guide, Elsevier, Morgan Kaufman publishers, 2008.
2.	Shibu K V, "Introduction to Embedded Systems", Tata McGraw Hill Education, Private Limited, 2nd Edition.
3.	Raghunandan...G.H, Microcontroller (ARM) and Embedded System, Cengage learning Publication, 2019
4.	The Insider's Guide to the ARM7 Based Microcontrollers, Hitex Ltd., 1st edition, 2005.
5.	Raj Kamal, Embedded System, Tata McGraw-Hill Publishers, 2nd Edition, 2008.

CO-PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	1	2	1	-	-	-	-	-	-	-	-
CO2	3	2	1	3	3	2	-	-	2	-	1	-
CO3	3	2	1	3	-	2	-	-	2	-	-	-
CO4	3	3	2	3	3	2	-	-	2	2	2	-
CO5	3	2	3	3	3	2	-	-	2	2	2	2

High-3, Medium-2, Low-1

Course Title	**Data Communication**	Semester	03
Course Code	**MVJ19CS46**	**CIE**	50
Total No. of Contact Hours	50 L : T : P :: 30 : 10 : 10	**SEE**	50
No. of Contact Hours/week	5	**Total**	100
Credits	3	**Exam. Duration**	3

Course objective is to:

- Define the protocol layering and physical level communication.
- To demonstrate the performance of a network.
- To be familiar the various components required to build different networks.
- To learn the functions of transport layer protocols to provide reliable communication.
- To familiarize the protocols of the Application layer.

Module-1	**RBT Level** L1, L2,L3	10 Hrs.

INTRODUCTION AND PHYSICAL LAYER

Introduction: Networks – Network Types – Protocols and Standards– TCP/IP Protocol suite – OSI Model, Data Encoding: Line Encoding-Types of Line Coding- Analog-to-Digital Conversion- Pulse code modulation (PCM)-Delta modulation (DM)-Transmission Modes.

Laboratory Sessions/ Experimental learning: Design the simulation system for performing analog to digital conversion.

Applications: Mobile Phone, Laptop and all electronic devices

Video link / Additional online information (related to module if any): https://www.digimat.in/nptel/courses/video/106105183/L01.html

Module-2	**RBT Level** L1, L2	10 Hrs.

DATA-LINK LAYER

Introduction – Link-Layer Addressing – DLC Services – Error Detection and Correction: Introduction, Block coding, Cyclic codes, Checksum. Error Correction and Detection Protocols: Protocols for Noiseless Channels- Simplest protocol, Stop-and-wait protocol; Protocols for Noisy

Channels- Stop-and-wait automatic repeat request, Go − back − N − automatic repeat request, Selective repeat automatic repeat request.

Laboratory Sessions/ Experimental learning: Develop the system for error correction code (like CRC) and verify the reliability of data at both sides.

Applications: Telecommunication

Video link / Additional online information (related to module if any):

https://www.youtube.com/watch?v=pVl1L1jrbFE

Module-3	**RBT Level** L1, L2,L3	10 Hrs.

MEDIA ACCESS CONTROL

Media Access control: Random Access, Controlled Access and Channelization, Wired LANs Ethernet: Ethernet Protocol, Standard Ethernet, Fast Ethernet, Gigabit Ethernet and 10 Gigabit Ethernet, Wireless LANs: Introduction, IEEE 802.11 Project and Bluetooth.

Laboratory Sessions/ Experimental learning: Create the virtual environment for WLAN and make the data communication between stations.

Applications: Making communication between devices

Video link / Additional online information (related to module if any):

https://www.youtube.com/watch?v=5u52wbqBgEY

Module-4	**RBT Level** L1, L2,L3	10 Hrs.

NETWORK LAYER

Network Layer Services − Packet switching − Performance − IPV4 Addresses − Forwarding of IP Packets, IP Addressing Scheme- Subnet Addressing-Subnet Masks-IPV4 Addressing-IPV6 Addressing- Address Resolution Protocol (ARP)-Reverse Address Resolution Protocol (RARP)

Laboratory Sessions/ Experimental learning: Write a code finding the physical address and logical address of the system using ARP /RARP protocols.

Applications: Resolve addressing problem in systems

Video link / Additional online information (related to module if any):

https://www.youtube.com/watch?v=rW1jPlYgp_0

Module-5	**RBT Level** L1, L2,L3	10 Hrs.

TRANSPORT LAYER

Introduction − Services of Transport Layer, Connection Establishment, Connection Release, Transport Layer Protocols- TCP protocol, UDP protocol; Congestion: TCP Congestion control − Congestion avoidance (DECbit, RED)

Laboratory Sessions/ Experimental learning: Create the system for avoiding congestion in

unreliable communication.

Applications: Reliable communication among devices in network like LAN,WAN etc.

Video link / Additional online information (related to module if any): https://www.youtube.com/watch?v=z_lCsUGwr3U

Course outcomes:	
CO1	Identify the components required to build different types of networks.
CO2	Choose the required functionality at each layer for given application
CO3	Identify solutions for each functionality at each layer
CO4	Trace the flow of information from one node to another node in the network.
CO5	Analyse the working of various application layer protocols

Reference Books:	
1.	Behrouz A. Forouzan, Data Communications and Networking 5E, 5th Edition, Tata McGraw-Hill, 2013.
2.	Larry L. Peterson and Bruce S. Davie: Computer Networks − A Systems Approach, 5th Edition, Morgan Kaufmann Publishers Inc, 2012.
3.	William Stallings, Data and computer communication Networks, Second edition, Pearson education, 2013.
4.	Ying-Dar Lin, Ren-Hung Hwang, Fred Baker, ″Computer Networks: An Open Source Approach″,Mc Graw Hill Publisher, 2011.

CO-PO Mapping												
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	2	1	-	-	-	-	-	-	2	-
CO2	2	2	2	1	-	-	-	-	-	-	2	1
CO3	2	2	2	1	-	-	-	-	-	-	2	-
CO4	1	2	2	1	-	-	-	-	-	-	2	1
CO5	1	2	2	1	-	-	-	-	-	-	2	2

High-3, Medium-2, Low-1

Course Title	**Analysis and Design of Algorithms- Lab**	Semester	04
Course Code	**MVJ19CSL47**	**CIE**	50
Total No. of Contact Hours	40 L : T : P :: 10 : 10 : 20	**SEE**	50

No. of Contact Hours/week	3	**Total**	100
Credits	2	**Exam. Duration**	3

Course objective is to:

- To employ various design strategies for problem solving.
- To provide exposure to measure and compare the performance of different algorithms.
- To provide design and implement various Concepts in JAVA.

Laboratory Sessions

Sl.No	**Experiment**	**RBT Level**
1.	Write a recursive program to a. Solve Towers-of-Hanoi problem b.GCD	**L3**
2.	Write a Java program to implement the Stack using arrays. Write Push(), Pop(), and Display() methods to demonstrate its working.	**L3**
3.	Implement Recursive Binary search and Linear search and determine the time required to search an element. Repeat the experiment for different values of N and plot a graph of the time taken versus N.	**L3**
4.	Given a set of N integer elements which is to be sorted using Selection Sort technique. Write the program using C language as well as in Java for different values of N and observe the total time taken to sort the elements in both the languages. .	**L3**
5.	Write program to do the following: a. Print all the nodes reachable from a given starting node in a digraph using BFS method. b. Check whether a given graph is connected or not using DFS method	**L3**
6.	The Merge sort is one of the most common algorithms used to sort arrays. The class Merge sort implements this algorithm. However, there is a bug in the implementation of the method sort. Debug the previous implementation using the	**L3**

	debugging options of your favourite IDE (e.g. eclipse), in order to find the error.	
7.	Sort a given set of N integer elements using Quick Sort technique and Run the program for different values of N and record the time taken to sort.	**L3**
8.	We are given a set of items, each with a weight and a value and we need to determine the number of each items to include in a collection so that the total weight is less than or equal to the given limit and the total value is as large as possible. Write a Java program by applying any reuse sub problem technique to find the solution.	**L3**
9.	Suppose you're trying to find the shortest path from your house to various locations like Movie theatre, Gas Station,Grocery Store and Petrol pump. If we let various locations be vertices and the routes between them are edges, we can create a weighted graph representing the situation. Write a Java program to find the shortest path from your house (source) to the remaining locations.	**L3**
10.	Write a Java program for the following Scenario, You have a business with several offices and you want to lease phone lines to connect them up with each other; and the phone company charges different amounts of money to connect different pairs of cities. You want a set of lines that connects all your offices with a minimum total cost and it should be a spanning tree.	**L3**
11.	Develop a program in Java with a given set of vertices V in a weighted graph where each edge w (u,v) can be negative, find the shortest path weights d(s,v) from every source s to all vertices in the graph. If the graph contains negative cycle, report it.	**L3**
12.	Given a set of cities and distance between every pair of cities, the problem is to find the shortest possible route that visits every city exactly once and returns to the starting point. Write a program to find the solution using dynamic programming method.	**L3**

13.	Given a set of positive integers and an integer ‘s' write a program in Java to determine whether there is any non-empty subset whose sum is ‘s'.	**L3**
14.	Write a Java program to find a path that traverses all the vertices of the given graph G exactly once and then ends at the starting vertex in a connected undirected Graph G of *n* vertices using backtracking principle	**L3**

Course outcomes:	
CO1	Design algorithms using appropriate design techniques (brute-force, greedy, dynamic programming, etc.)
CO2	Implement a variety of algorithms such as sorting, graph related, combinatorial, etc., in a high level language.
CO3	Analyze and compare the performance of algorithms using language features
CO4	Apply and implement learned algorithm design techniques and data structures to solve real-world problems.
CO5	Employ various design strategies for problem solving and implement various algorithms in JAVA .

CO-PO Mapping												
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	-	-	-	-	-	3	-	2	2
CO2	3	3	2	-	-	-	-	-	3	-	2	2
CO3	3	3	2	-	-	-	-	-	3	-	2	2
CO4	3	3	2	-	-	-	-	-	3	-	2	2
CO5	3	3	2	-	-	-	-	-	3	-	2	2

High-3, Medium-2, Low-1

Course Title	**Micro Controller and Embeded Systems Lab**	Semester	IV
Course Code	**MVJ19CSL48**	**CIE**	50
Total No. of Contact Hours	40 L : T : P :: 10 : 10 : 20	**SEE**	50
No. of Contact Hours/week	3	**Total**	100
Credits	2	**Exam. Duration**	3 HOURS

Course objective:

This course will enable students to get practical experience in design, develop, implement, analyze and evaluation of

- Linear data structures and their applications such as stacks, queues and lists,
- Non-Linear data structures and their applications such as Trees &Graphs
- Sorting and Hashing techniques

Sl.No	Experiment	RBT Level
1	Write a program to find the sum of first 10 integer numbers.	L3
2	Write a program to find factorial of a number.	L3
3	Write a program to add an array of 16 bit numbers and store the 32 bit result in internal RAM	L3
4	Write a program to find the square of a number (1 to 10) using look-up table.	L3
5	Write a program to find the largest/smallest number in an array of 32 numbers	L3
6	Write a program to arrange a series of 32 bit numbers in ascending/descending order	L3
7	Write a program to count the number of ones and zeros in two consecutive memory locations	L3
8	Write an ARM assembly program that checks if a 32-bit number is a palindrome. Assume that the input is available in r 3. The program should set r 4 to 1 if it is a palindrome, otherwise r 4 should have 0. A palindrome is a number which is the same when read from both sides. For example, 1001 is a 4 bit palindrome.	L3
9	Display “Hello World” message using Internal UART	L3
10	Interface and Control a DC Motor	L3
11	Interface a Stepper motor and rotate it in clockwise and anti-clockwise direction	L3
12	Interface a DAC and generate Triangular and Square waveforms.	L3
13	Display the Hex digits 0 to F on a 7-segment LED interface, with an appropriate delay in Between	L3
	STUDY EXPERIMENT	L3

1.	Interface a 4x4 keyboard and display the key code on an LCD	

Course outcomes:	
CO1	Analyze and Compare various linear data structures
CO2	Code, debug and demonstrate the working nature of different types of data structures and their applications
CO3	Implement, analyse and evaluate the searching and sorting algorithms
CO4	Choose the appropriate data structure for solving real world problems

---

CO-PO Mapping												
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	-	3	3	-	-	3	-	3	2
CO2	3	3	2	-	3	3	-	-	3	-	3	2
CO3	3	3	2	-	3	3	-	-	3	-	3	2
CO4	3	3	2	-	3	3	-	-	3	-	3	2

High-3, Medium-2, Low-1