MIT Academy of Engineering (An Autonomous Institute Affiliated to SPPU)	COURSE SYLLABI (2019 – 2023)	
SCHOOL OF COMPUTER ENGINEERING AND TECHNOLOGY	**W.E.F**	**AY:** 2020 - 2021
SECOND YEAR BACHELOR OF TECHNOLOGY INFORMATION TECHNOLOGY	**COURSE NAME**	Advanced Data Structures
	COURSE CODE	CS228
	COURSE CREDITS	4
RELEASED DATE : 01/07/2020	**REVISION NO**	1.0

TEACHING SCHEME (HOURS/WEEK)		EXAMINATION SCHEME & MARKS						
		THEORY			PRACTICAL			TOTAL
LECTURE	PRACTICAL	MSE	ESE	IA	MSE	ESE	IA	
3	2	35	35	30	NIL	40	10	150

PRE-REQUISITE:

1: Data Structures
2: Discrete Structures and Graph Theory

COURSE OBJECTIVES:

CS228.CEO.1: Introduce various advanced data structures like trees, graphs, heaps, hash tables, disjoint sets etc.
CS228.CEO.2: Learn how to select appropriate data structure based on requirement of application
CS228.CEO.3: Learn how to implement various applications using data structures
CS228.CEO.4: Learn how to measure the performance of data structure in terms of time and memory complexity
CS228.CEO.5: Learn how to design own data structure using standard data structure

COURSE OUTCOMES:

The students after completion of the course will be able to,

CS228.CO.1: Explain the working of basic and advanced data structures like trees, graphs, heaps, disjoint sets, hash tables, bloom filters
CS228.CO.2: Demonstrate the advantages and disadvantages of various data structures
CS228.CO.3: Choose appropriate data structures while building the applications
CS228.CO.4: Implement various applications using data structures like trees, graphs, hash tables, heaps
CS228.CO.5: Evaluate the performance of various data structures in terms of time and memory complexity
CS228.CO.6: Design own data structures using the build in data structures

THEORY COURSE CONTENT		
UNIT 1	**Trees**	**8 HOURS**

App/System/Case study: Parse trees and expression trees in Compiler
Contents: General tree and its representation using sequential and linked organization, converting tree to binary tree, binary tree traversals – inorder, preorder and postorder traversals, breadth first search traversal, binary tree operations. Binary Search Tree (BST), operations on BST. Threaded Binary Tree – concept, threading, insertion and deletion of nodes in threaded binary tree, inorder, preorder and postorder traversals of threaded binary tree.
Self study: Game trees
Further Reading: Optimal Binary Search Tree

UNIT 2	**Height Balanced and Multiway Trees**	**9 HOURS**

App/System/Case study: BTRFS File System
Contents: AVL Trees, B Trees, B+ Trees, Trie Trees, Splay Trees, Red Black Trees
Self study: AA Trees
Further Reading: Range queries

UNIT 3	**Graphs**	**6 HOURS**

App/System/Case study: Packet routing in networks
Contents: Basic Concepts, Storage representation, Adjacency matrix, adjacency list, adjacency multi list, inverse adjacency list. Traversals - depth first and breadth first, Minimum spanning Tree, Prims and Kruskal Algorithms, Dikjtra's Single source shortest path, Topological ordering
Self Study: Warshall's algorithm
Further Reading: Algorithms for connected components

UNIT 4	**Heaps and Disjoint Sets**	**6 HOURS**

App/System/Case study: Priority queue
Contents: Concept of Min and Max Heap, Operations on Heap – insert, delete, up-heapify and down-heapify, use of heap in heap-sort.
Concept of Disjoint Sets, Disjoint Sets as ADT, Up Trees, Smart Union and Path Compression
Self Study: Fibonacci heaps
Further Reading: Min - Max heaps, Multidimensional Heaps

UNIT 5	**Hashing**	**6 HOURS**

App/System/Case study: Cryptographic hash functions
Contents: Concepts - hash table, hash function, bucket, collision, probe, overflow, open hashing, closed hashing, perfect hash function, load density, full table, load factor, rehashing. hash functions - properties of good hash function, division, multiplication, extraction, mid - square, folding and universal hash function, Collision resolution strategies - open addressing and chaining
Self Study: Extendable hashing
Further Reading: Locality sensitive hashing

UNIT 6	Probablisitc Data Structures	6 HOURS

App/System/Case study: Recommendation System

Contents: Bloom Filters, Cuckoo Filters, Quotient Filters, Count-Min Sketch, HyperLogLog, Minhash, Simhash

Self Study: Counting Bloom Filter

Further Reading: q-digest and t-digest data structures for ranking

PRACTICAL:		
PRACTICAL NO.01	**Binary Search Tree**	**4 HOURS**
Write a program in C++ to implement the following operations on Binary Search Tree: create, recursive inorder traversal, recursive preorder traversal, recursive postorder traversal, non recursive inorder traversal, non recursive preorder traversal, non recursive postorder traversal, delete a node, insert a node, level wise printing		
PRACTICAL NO.02	**Threaded Binary Tree**	**4 HOURS**
Write a program in C++ to implement the following operations on Threaded Binary Tree : create, recursive inorder traversal, recursive preorder traversal, recursive postorder traversal, non recursive inorder traversal, non recursive preorder traversal, non recursive postorder traversal		
PRACTICAL NO.03	**AVL Tree**	**4 HOURS**
Write a program in C++ to implement the following operations on AVL Trees: create, recursive inorder traversal, recursive preorder traversal, recursive postorder traversal, non recursive inorder traversal, non recursive preorder traversal, non recursive postorder traversal		
PRACTICAL NO.04	**Minimum Spanning Tree**	**4 HOURS**
Write a program in C++ to find the minimum spanning tree of a given graph using Prim's and Kruskal's algorithm		
PRACTICAL NO.05	**Hashing**	**4 HOURS**
Write a program in C++ to implement hash tables with collision handling strategies		
PRACTICAL NO.06	**Heaps**	**4 HOURS**
Write a program in C++ to implement heap sort algorithm to sort the given numbers in ascending and descending order		
PRACTICAL NO.07	**Probabilistic Data Structures**	**4 HOURS**
Write a program in C++ to implement the following probabilistic data structures: Bloom's Filter, Cuckoo Filter		

TEXT BOOK

1. Horowitz, Ellis, Sartaj Sahni, and Susan Anderson-Freed. Fundamentals of data structures. Vol. 1982. Potomac, MD: Computer science press, 1976
2. Samanta, Debasis. Classic data structures. Vol. 2. Prentice Hall India, 2001.
3. Brass, Peter. Advanced data structures. Vol. 193. Cambridge: Cambridge University Press, 2008

REFERENCE BOOK

1. Cormen, Thomas H., et al. Introduction to algorithms. MIT press, 2009
2. Gakhov, Andrii. Probabilistic Data Structures and Algorithms for Big Data Applications, 2019.
3. Mehta, Dinesh P., and Sartaj Sahni. Handbook of data structures and applications. Chapman and Hall/CRC, 2004.