Course code	Course Name	L-T-P-Credits	Year of Introduction
CS482	DATA STRUCTURES	3-0-0-3	2016

Pre-requisite: A course on C or C++ in the B-Tech level with emphasis on pointers and functions.

Course Objectives:

- To introduce linear data structures such as stacks, queues and their applications.
- To introduce non-linear data structures such as trees, graphs and their applications.
- To impart various sorting, searching and hashing techniques and their performance comparison.

Syllabus:

Introduction to various programming methodologies, terminologies and basics of algorithms analysis, Basic Abstract and Concrete Linear Data Structures, Non-linear Data Structures, Sorting Algorithms, Searching Algorithms, Hashing.

Expected Outcome:

The Student will be able to:-

i. compare different programming methodologies and define asymptotic notations to analyze performance of algorithms
ii. choose appropriate data structures like arrays, linked list, stacks and queues to for practical scenarios
iii. represent and manipulate data using nonlinear data structures like trees and graphs to design algorithms for various applications
iv. illustrate and compare various techniques for searching and sorting
v. illustrate various hashing techniques

Text Books:

1. Richard F. Gilberg and Behrouz A. Forouzan, Data Structures: A Pseudocode Approach with C, 2/e, Cengage Learning, 2005.
2. Samanta D., Classic Data Structures, Prentice Hall India, 2/e, 2009.

References:

1. Aho A. V., J. E. Hopcroft and J. D. Ullman, Data Structures and Algorithms, Pearson Publication,1983.
2. Horwitz E., S. Sahni and S. Anderson, Fundamentals of Data Structures in C, University Press (India), 2008.
3. Hugges J. K. and J. I. Michtm, A Structured Approach to Programming, PHI, 1987.
4. Lipschuts S., Theory and Problems of Data Structures, Schaum's Series, 1986.
5. Martin Barrett, Clifford Wagner, And Unix: Tools For Software Design, John Wiley, 2008
6. Peter Brass, Advanced Data Structures, Cambridge University Press, 2008
7. Tremblay J. P. and P. G. Sorenson, Introduction to Data Structures with Applications, Tata McGraw Hill, 1995.
8. Wirth N., Algorithms + Data Structures = Programs, Prentice Hall, 2004.

COURSE PLAN

Module	Contents	Hours	End Sem. Exam Marks

I	Introduction to programming methodologies – structured approach, stepwise refinement techniques, programming style, documentation – analysis of algorithms: frequency count, definition of O notation, asymptotic analysis of simple algorithms. Recursive and iterative algorithms.	**06**	**15%**
II	Abstract and Concrete Data Structures- Basic data structures – Arrays, Linked lists:- singly linked list, doubly linked list, Circular linked list, operations on linked list, linked list with header nodes, applications of linked list: polynomials,.	**07**	**15%**
FIRST INTERNAL EXAMINATION			
III	Implementation of Stacks and Queues using arrays and linked lists, Applications. Trees: - m-ary Tree, Binary Trees – level and height of the tree, complete-binary tree representation using array, tree traversals (Recursive only), applications.	**07**	**15%**
IV	Binary search tree – creation, insertion and deletion and search operations, applications. Heaps- Min-max heaps, Graphs – representation of graphs, BFS and DFS (analysis not required) applications.	**06**	**15%**
SECOND INTERNAL EXAMINATION			
V	Minimum Spanning Trees – Prim's and Kruskal algorithms. Shortest path algorithms – Djikstra and Warshall algorithms Sorting techniques – Bubble sort, Selection Sort, Insertion sort, Merge sort, Quick sort, Searching algorithms (Performance comparison expected. Detailed analysis not required)	**07**	**20%**
VI	Linear and Binary search. (Performance comparison expected. Detailed analysis not required) Hash Tables – Hashing functions – Mid square, division, folding, digit analysis, collusion resolution and Overflow handling techniques.	**07**	**20%**
END SEMESTER EXAMINATION			

Question Paper Pattern (End semester exam)

1. There will be ***FOUR*** parts in the question paper – **A, B, C, D**
2. **Part A**
   a. **Total marks : 40**
   b. ***TEN*** questions, each have **4 marks**, covering **all the SIX modules** (***THREE*** questions from **modules I & II**; ***THREE*** questions from **modules III & IV**; ***FOUR*** questions from **modules V & VI**).
   All the TEN questions have to be answered.
3. **Part B**
   a. **Total marks : 18**
   b. ***THREE*** questions, each having **9 marks**. One question is from **module I**; one question is from **module II**; one question ***uniformly*** covers **modules I & II**.
   c. ***Any TWO*** questions have to be answered.
   d. Each question can have ***maximum THREE*** subparts.

4. **Part C**
    a. **Total marks : 18**
    b. ***THREE*** questions, each having **9 marks**. One question is from **module III**; one question is from **module IV**; one question ***uniformly*** covers **modules III & IV**.
    c. ***Any TWO*** questions have to be answered.
    d. Each question can have ***maximum THREE*** subparts.
5. **Part D**
    a. **Total marks : 24**
    b. ***THREE*** questions, each having **12 marks**. One question is from **module V**; one question is from **module VI**; one question ***uniformly*** covers **modules V & VI**.
    c. ***Any TWO*** questions have to be answered.
    d. Each question can have ***maximum THREE*** subparts.
6. There will be ***AT LEAST* 60%** analytical/programming/numerical questions in all possible combinations of question choices.