

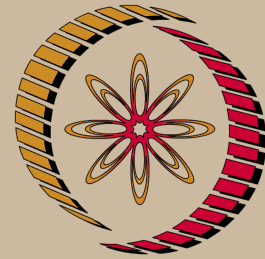
Programming and Data structures (PDS) - Video course

COURSE OUTLINE

The focus of the course is program design using data structures. After this course, the student will be able to analyze the difference between the choice of different data structures for a given programming task. The student will be able to write programs involving different data structures, and also appreciate the value of asymptotic analysis of data structure set-up times, maintenance times and space used.

COURSE DETAIL

Week. No	Topics
1.	Introduction to Abstract Data Types and analysis of different algorithms <ul style="list-style-type: none"> Review of elementary data types and structures in C. The Array data type and the importance of Random Access. Searching an array: linear and Binary search. Sorting: Merge Sort, and analysis
2.	ADT Array -- searching and sorting on arrays. <ul style="list-style-type: none"> Review of Pointers in C. The Linked list ADT. Searching a linked list, inserting and deleting from a linked list. Application: representing a univariate polynomial, and adding two univariate polynomials
3.	ADT Linked Lists, Stacks, Queues. <ul style="list-style-type: none"> List manipulation algorithms: reversal of a list, use of recursion to reverse/search. Doubly linked lists, circular linked lists.



NP-TEL

NPTEL

<http://nptel.ac.in>

Computer
Science and
Engineering

Coordinators:

Narayanaswamy N S

Associate Professor
Department of
Computer Science Engineering
IIT Madras

	<ul style="list-style-type: none"> Stack and Queue ADT, comparison of implementation using arrays and linked lists
4.	<p>Binary Trees</p> <ul style="list-style-type: none"> Tree ADT representation, traversal, application of binary trees in Huffman coding. Introduction to expression trees: Recursive traversal depth, height, and number of nodes. post/pre/infix notation.
5.	<p>Dictionary</p> <ul style="list-style-type: none"> Binary search trees search, insertion and deletion Balanced binary search trees.
6.	<p>ADT Priority queues</p> <ul style="list-style-type: none"> Heap ADT implementation and Heapsort, in place sorting. Heaps for maintaining interval trees.
7.	<p>Graphs</p> <ul style="list-style-type: none"> Representations or relations using matrices. The Graph ADT and applications Transitive closure, Floyd Warshall's algorithm and applications connectivity and spanning trees.
8.	<p>Advanced topics options for the teacher</p> <ul style="list-style-type: none"> Adj. List representation of a Graph. Breadth First Search traversal and identification of shortest paths. Depth First Search recursive specification and application to finding articulation points.