

# IC250 Data Structure and Algorithms

---

Credit: 1-0-3-3

Prerequisite: Consent of the faculty member

Students intended for: B.Tech

Elective or Core: Core

Semester: Even/Odd

## Course objective:

This is an introductory-level course in Data structures and algorithms offered to all engineering disciplines. The main objective of the course is to educate students about various useful data structures and algorithms, how to implement them and why a particular data structure or algorithm is better than other. Emphasis is on problem solving using various data structures and algorithms.

## Course Contents:

- Introduction to data structures, abstract data types, Creation and manipulation of data structures: arrays, lists, stacks, queues, trees, heaps, hash tables, balanced trees. Algorithms for sorting and searching. Notion of time and space complexity, the O- notation
- Introduction: Role of algorithms in Computing, analysing algorithms and designing algorithms [2 Lectures]
- Data Structures: Stacks, queues, linked lists, rooted tress, B-tree, graphs, hash tables, recursion [6 Lectures]
- Sorting and searching algorithms: Bubble, Heapsort, Quicksort, Sequential Searches, Binary search [4 Lectures]
- Complexity: Time and Space complexity, O-Notation, P vs NP, NP hard vs NP complete [2 Lectures]

## Lab Exercises:

- Lab to be conducted on a 3-hour slot. It will be conducted in tandem with the theory course so the topics for problems given in the lab are already initiated in the theory class. The topics taught in the theory course should be appropriately be sequenced for synchronization with the laboratory. A sample sequence of topics and lab classes for the topic are given below
- Two assignments: Designing algorithm for some problems and writing program for it
- Four-Five assignments (some examples: Building a queue of strings, practice with linked data structures, Using a stack to evaluate arithmetic expressions etc),
- Two-Three assignments: sorting with recursion etc
- Four-Five assignments: Comparing time and space complexity, e.g, comparing sorting by minimum search and sorting by mergesort, Analysing NP hard and NP complete problems and dealing with them

## Text Book:

Introduction to Algorithms-Thomas H.Cormen, Leiserson, Rivest, Stein (Text Book)

## References

Data Structures and Algorithms, By: Alfred V. Aho, Jeffrey D. Ullman, John E. Hopcroft