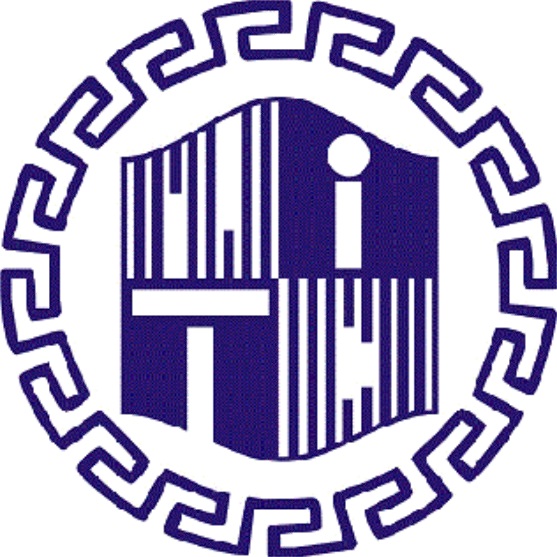
**NATIONAL INSTITUTE OF TECHNOLOGY,**

**DELHI**

ASSIGNMENT -5



**DESIGN & ANALYSIS OF**

**ALGORITHMS ( CSB -252 )**

**Name-Abhishek Luthra Branch – CSE 2nd Year**

**Roll no.- 181210003**

**Deterministic Polynomial Time (P):**

This is the class of problems which contains all the problems that can be solved by a deterministic Turing machine within a polynomial computation time. Examples for such problems are searching algorithms, sorting algorithms, etc. which can all be done in polynomial time.

**Non-Deterministic Polynomial Time (NP):**

NP is the set of decision problems that are solvable in polynomial time by a Non-deterministic Turing Machine. NP problems do not have a know algorithm that can produce a result in polynomial time. If we are given a solution to an NP problem, verifying that it is correct is easy and can be done in polynomial time or less. They are easy to check but hard to solve, they generally have an exponential time complexity such as O(nn).

**NP – Hard:**

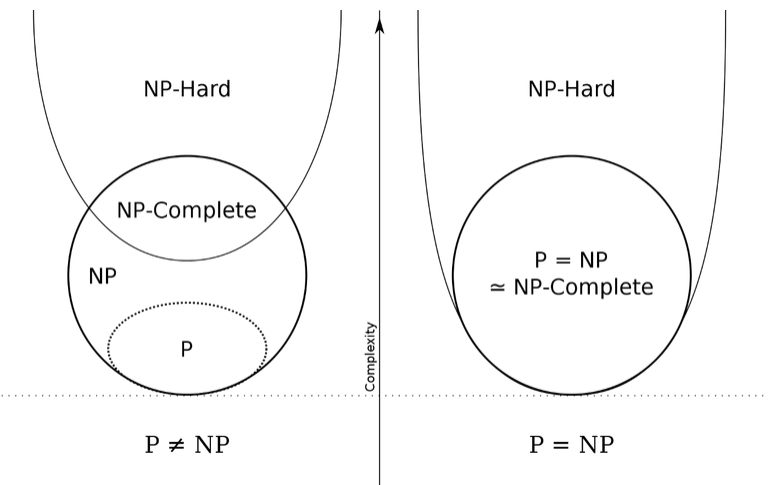
This is the class of problems which are at least as hard as the hardest problems in NP. Problems belonging to this class may or may not be part of NP, that is, the hardest problems of NP belong to the intersection of NP and NP-Hard. Problems in NP-Hard may not even be decision problems.

Example of a problem which is NP-Hard but not NP is the problem of identifying a chess move in any given board state that is the best possible move to make.

**NP – Complete:**

This is the class problems which contains the set of all the hardest problems in NP. Every problem in NP-Complete must belong to NP and NP-Hard, which is not true for NP-Hard. NP-Complete is the intersection of NP and NP-Hard.

Example of a problem which is NP-Complete is the clique graph problem, where, in an undirected graph, the largest complete sub-graph is to be found.



|  |  |  |
| --- | --- | --- |
| Sr No. | P Problems | NP Problems |
| 1. | These can be solved in polynomial time by deterministic algorithms. | These can be solved in non-deterministic polynomial time. |
| 2. | Such problems can be solved and verified in polynomial time. | NP problems solution cannot be obtained in polynomial time but if solution is given it can be verified in polynomial time. |
| 3. | P problems are subset of NP problems. | NP problems are a superset of P problems. |
| 4. | Ex. Searching, Sorting, Addition, Multiplication, etc. | Ex. Sudoku, Travelling Salesperson, Knapsack Problem |