Name: Keyur Patel
Roll no: 1601042107 3
Batch: A2

Tut - 9

Explain
i) Decidability
ii) Classes P & NP

Ans(i) Decidability refers to the property of a problem or a language being solvable by a computes algorithm. In other words, a problem or language is decidable if there exists an algorithm that when given an input, will always terminate and correctly determine whether the input belongs to the language or not. If such an algorithm exists, the problem or language is said to be decidable.

For Eg: The problem of determining whether a given integer is even or odd is decidable.
You can write a simple Algorithm to check if the last digit of the integer is even or odd.
This algorithm will always terminate and correctly decide whether the integer is even or odd, making it decidable problem.

Ans (ii) Classes P & NP.

- P (Polynomial time):

P is a class of decision problems (problems with yes/no answer) that can be solved by a deterministic Turing machine in Polynomial time. In other words, problems in P can be solved efficiently with an

algorithm whose running time is bounded by a polynomial in the size of input.

Eg: The problem of checking whether a given list of numbers is sorted in ascending orders or not. You can solve this problem in O(n) time, where "n" is the number of elements in the list, by iterating through the list once.

- **N.P. (Non-Deterministic Polynomial time)**

NP is a class of decision problems for which a proposed solution can be verified in polynomial time. While finding solution might be difficult, if you're given a solution, you can efficiently check whether it's correct.

Eg: TSP (Travelling Salesperson Problem) is a classic NP Problem. Given a list of cities and the distances between them, finding the shortest route that visits each city exactly once is a difficult problem. However, if someone provides you with a route you can quickly calculate the total distance of that route to verify its optimality, and this verification can be done in Polynomial time.