Decision Problems, Decidability, and Complexity Classes

1. What is a decision problem?

A decision problem is a problem with a yes/no answer, framed as a question about membership in a set. In computer science, we usually describe it as: given an input string, does it belong to some language L?

Example: “Is the number n prime?” answer : yes or no.

Example: “Does this graph contain a Hamiltonian cycle?” answer : yes or no.

Decision problems are fundamental because they allow us to compare the difficulty of different problems in a uniform way.

2. What does it mean for a decision problem to be decidable?

A decision problem is decidable if there exists an algorithm that:

1. Always produces the correct yes/no answer, and
2. Always halts in finite time.

If no such algorithm exists, the problem is undecidable.

- Example of a decidable problem: primality testing.
- Example of an undecidable problem: the Halting Problem (no algorithm can determine, for all programs and inputs, whether the program will halt).

3. What is the class P?

- P stands for Polynomial time.
- It is the class of decision problems that can be solved efficiently, i.e., by an algorithm whose running time is bounded by a polynomial in the size of the input.
- These are the problems we think of as “tractable” or realistically computable.
- Examples: sorting, shortest path (Dijkstra’s algorithm), maximum flow.

4. What is the class NP?

- NP stands for Nondeterministic Polynomial time.
- It is the class of decision problems where, given a proposed solution, we can verify its correctness in polynomial time.

- Intuitively: maybe finding the solution is hard, but checking one is easy.
- Examples: Boolean satisfiability (SAT), traveling salesman decision problem, graph coloring.

5. What is the intuitive meaning of the “P versus NP” question?

The central open problem in computer science:

- P = NP?
- Put simply: if you can quickly verify a solution, can you also quickly find one?
- If P = NP, then many hard problems (like SAT, scheduling, optimization puzzles) could be solved efficiently.
- If P ≠ NP, then some problems will always resist efficient solving, even though checking solutions is easy.

6. If you resolve the P versus NP question, how much richer will you be?

- The Clay Mathematics Institute has designated it as one of the seven Millennium Prize Problems, with a reward of $1 million for a correct proof.
- Beyond money:
- If P = NP, the impact would be revolutionary — cryptography would collapse, optimization and AI would change radically.
- If P ≠ NP, it confirms the boundaries of efficient computation and secures the foundations of cryptography.

Conclusion:

Decision problems provide a clear framework for studying computational limits. Decidability tells us whether problems are solvable at all, while complexity classes P and NP help us measure efficiency. The “P vs NP” question is not just technical it strikes at the heart of what can and cannot be computed efficiently, with enormous theoretical and practical consequences.