MTech CSE – 1st Semester
Student ID: **A125023**
Student Name: **SRIJITA VERMA**

Question

Define the class **Co-NP**. Explain the type of problems that belong to this complexity class.

Answer

Preliminaries: Decision Problems and NP

In computational complexity theory, we primarily study *decision problems*, whose answers are either YES or NO.

Recall that a decision problem L belongs to the class **NP** if:

- for every input $x \in L$ (a YES-instance),
- there exists a certificate (or witness) that can be verified in polynomial time by a deterministic Turing machine.

Thus, **NP** focuses on problems where YES answers are efficiently verifiable.

Definition of the Class Co-NP

Formal Definition

A decision problem L belongs to the class **Co-NP** if and only if its complement belongs to **NP**.

Formally,

$$L \in \textbf{Co-NP} \quad \Longleftrightarrow \quad \overline{L} \in \textbf{NP},$$

where

$$\overline{L} = \{\, x \mid x \notin L \,\}.$$

Equivalent Interpretation

A problem is in **Co-NP** if:

- for every NO-instance, there exists a certificate,
- that can be verified in polynomial time.

In other words, **Co-NP** is the class of problems whose NO answers are efficiently verifiable.

Intuition Behind Co-NP

- **NP** problems: *"If the answer is YES, I can quickly verify why."*
- **Co-NP** problems: *"If the answer is NO, I can quickly verify why."*

Co-NP captures problems where it is difficult to prove that something exists, but easy to prove that it does not exist.

Relationship Between NP and Co-NP

Every problem in **P** belongs to both **NP** and **Co-NP**:

$$\mathbf{P} \subseteq \mathbf{NP} \cap \mathbf{Co\text{-}NP}.$$

It is unknown whether:

$$\mathbf{NP} = \mathbf{Co\text{-}NP}.$$

Most complexity theorists believe:

$$\mathbf{NP} \neq \mathbf{Co\text{-}NP},$$

although this has not been proven. This question is closely related to the famous **P** vs. **NP** problem.

Comparison of Complexity Classes: P, NP, and Co-NP

Class	Certificate Verified	Verification Time	Example Problem
P	YES or NO	Polynomial	Shortest Path
NP	YES instance	Polynomial	SAT
Co-NP	NO instance	Polynomial	UNSAT

Interpretation

- Problems in **P** can be solved directly in polynomial time.
- Problems in **NP** may be hard to solve, but YES answers are easy to verify.
- Problems in **Co-NP** may be hard to solve, but NO answers are easy to verify.

Types of Problems in Co-NP

Problems in **Co-NP** typically involve *universal claims*, such as:

- "No solution exists,"

- "Every possible configuration satisfies a property,"
- "A structure is valid for all cases."

These problems are often complements of well-known **NP** problems.

Canonical Examples of Co-NP Problems

TAUTOLOGY

Problem: Given a Boolean formula φ, is φ true under all truth assignments?

The complement of this problem is SAT. Since SAT is in **NP**, TAUTOLOGY is in **Co-NP** and is **Co-NP**-complete.

UNSAT (Unsatisfiability)

Problem: Given a Boolean formula φ, is it unsatisfiable?

Since UNSAT is the complement of SAT, and SAT is **NP**-complete, UNSAT is **Co-NP**-complete.

Graph Non-Hamiltonicity

Problem: Given a graph G, does G not contain a Hamiltonian cycle?

The Hamiltonian Cycle problem is **NP**-complete, hence its complement lies in **Co-NP**.

Composite Number Verification

Problem: Given a number n, is n composite?

This problem is in **NP** (a non-trivial factor is a certificate). Its complement, primality testing, lies in **Co-NP** and is also known to be in **P**.

Co-NP-Complete Problems

A problem L is **Co-NP**-complete if:

- $L \in$ **Co-NP**, and
- every problem in **Co-NP** can be reduced to L in polynomial time.

Examples include:

- TAUTOLOGY,

- UNSAT.

How Co-NP-Completeness Is Proved

To prove that a problem L is **Co-NP-complete**, two conditions must be satisfied:

1. Membership in Co-NP

One must show that the complement problem $\overline{L}$ belongs to NP. Equivalently, there must exist a polynomial-time verifiable certificate for NO-instances of L.

2. Co-NP-Hardness

One must show that every problem in Co-NP can be reduced to L in polynomial time.

In practice, this is done by:

- taking a known Co-NP-complete problem (such as UNSAT),
- giving a polynomial-time reduction from it to L.

Key Observation

Since Co-NP problems are complements of NP problems, Co-NP-completeness proofs often rely on known NP-completeness results by complementing both the problem and the reduction.

For example:

- SAT is NP-complete,
- UNSAT is therefore Co-NP-complete.

Structural Properties of Co-NP

- **Co-NP** is closed under complement by definition.
- It is not known to be closed under union or intersection.
- If **NP** = **Co-NP**, the polynomial hierarchy collapses.

Final Conclusion

Co-NP is the class of decision problems whose NO-instances have polynomial-time verifiable certificates.

In summary:

- **Co-NP** consists of complements of **NP** problems,
- it captures problems with efficiently verifiable non-existence proofs,
- it contains problems such as UNSAT and TAUTOLOGY,
- whether **Co-NP** = **NP** remains an open question.

Polynomial Hierarchy Remark. If NP = Co-NP, then the polynomial hierarchy collapses to its first level.

Intuition

NP verifies existence, whereas **Co-NP** verifies non-existence.