NP vs CO-NP

CO-NP:

Problem: Composite Numbers

Given an integer n, determine whether it's a composite number (not prime). In other words, the problem is to verify that the answer is "no" and provide evidence that n has divisors other than 1 and itself.

Verification for "No" Answer:

To verify that an integer n is not prime (i.e., the answer is "no"), you can check if it has divisors other than 1 and itself. If you find such divisors, you can efficiently conclude that n is composite.

Example:

Let's say n = 15. To verify that it's not prime, you can check if it has divisors other than 1 and 15. In this case, you can easily find divisors, such as 3 and 5. This verification process is efficient and confirms that the answer is "no" (15 is not prime).

This problem is in co-NP because it involves verifying the "no" answer (n is not prime) efficiently. If someone claims that an integer is not prime (answer "no"), you can efficiently check and confirm their claim by identifying divisors.

NP:

Problem: Hamiltonian Path

Given a directed or undirected graph G, is there a path that visits every vertex exactly once, and returns to the starting vertex? This is called a Hamiltonian Path.

Verification for "Yes" Answer:

If someone claims that a Hamiltonian path exists in a given graph, you can verify their claim efficiently by checking that the path they provide indeed visits each vertex once and returns to the starting vertex. This verification can be done in polynomial time.

Example:

Consider an undirected graph G with the following vertices and edges:

Vertices: A, B, C, D

Edges: (A, B), (B, C), (C, D), (D, A)

Claim: There is a Hamiltonian path in the graph.

You can verify this claim by examining a path that starts at one vertex (e.g., A), follows the edges (A -> B -> C -> D), and returns to the starting vertex (A). This path satisfies the Hamiltonian path condition, and you can verify it efficiently.

Since the verification of a "yes" answer (the existence of a Hamiltonian path) can be done in polynomial time, the Hamiltonian Path problem is in NP.

In co-NP, you would typically have problems where you efficiently verify a "no" answer. Problems like Prime Factorization is an example of a problem in NP where you efficiently verify a "yes" answer. If you find a number's prime factors, you can easily verify their correctness by multiplying them.