Project Overview: Currency Arbitrage Detection

Real-Time Story

Imagine you work as a quantitative trader at a financial institution. Your task is to monitor exchange rates among various currencies in real time. The trading system must quickly detect arbitrage opportunities so that profitable trades can be executed before market conditions change. You are given a set of currencies and the exchange rates between them. By modeling the exchange market as a directed graph—with vertices representing currencies and edges representing exchange rates —you can use the Bellman–Ford algorithm to determine if any cycle in the graph would yield a profit (i.e. the product of the rates in the cycle is greater than 1). To do this, you convert each rate (r) into a weight ($w = -\log(r)$). In this new graph, a negative cycle indicates an arbitrage opportunity.

Problem Statement

Given a list of currencies and exchange rates between them, determine whether there exists an arbitrage opportunity. An arbitrage opportunity exists if there is a cycle in the graph such that the product of the exchange rates in the cycle is greater than 1.

Input Format

- **First Line:** Two integers, `N` and `M`, where `N` is the number of currencies (nodes) and `M` is the number of available exchange rates (directed edges).
- **Next M Lines:** Each line contains two integers and one floating-point number:
    - `u` – source currency (node)
    - `v` – destination currency (node)
    - `r` – the exchange rate from currency `u` to currency `v`

Example:

```
3 3
1 2 0.9
2 3 0.9
3 1 1.3
```

Output Format

- A single line indicating whether an arbitrage opportunity was detected:
    - If yes, output:
      ```
      Arbitrage Opportunity Detected
      ```

- If no, output:

```
No Arbitrage Opportunity
```