Exercise Description

Objective:
Implement a service discovery mechanism. You will build a system where a client first queries a simple UDP "directory" server to find the address of a more complex TCP calculation server.

Scenario:
Imagine a large network where the powerful TCP calculator server might change its location (IP address or port) for maintenance or load balancing. Instead of hard-coding the calculator's address into every client, we use a lightweight, stable UDP server as a directory.

The client's process is:

1. Send a request to the well-known UDP address server, asking, "Where is the calculator service?"
2. The UDP server replies with the current IP and port of the TCP calculator.
3. The client then establishes a direct TCP connection to the calculator to perform its tasks.

This pattern is very common in distributed systems, from online gaming (finding a match server) to microservices architecture.

Architecture:

1. **Client -> UDP Address Server:** `b"Hello Server"`
2. **UDP Address Server -> Client:** `"ip:port"` (e.g., `"127.0.0.1:10002"`)
3. **Client -> TCP Calc Server:** (Establishes TCP connection to the received address)
4. **Client <-> TCP Calc Server:** (Sends math problems and receives results)

Your Task:
Complete the `address_calc_client_exercise.py`, `address_calc_server_exercise.py`, and `calc_server_exercise.py` files by filling in the code marked with `TODO` comments.

Instructions for Running

You will need **three separate terminal windows**. The startup order is very important!

1. Terminal 1: Start the TCP Calculator Server

This is the final destination, so it must be running first. Let's run it on port `10002`.

```
python calc_server_exercise.py localhost 10002
```

You should see a message that it's listening.

2. Terminal 2: Start the UDP Address Server

This server needs to know the address of the TCP server you just started. Let's run it on port `10001`.

```
python address_calc_server_exercise.py localhost 10001 localhost:10002
```

It will now be listening for requests.

3. Terminal 3: Start the Client

The client only needs to know the address of the UDP directory server.

```
python address_calc_client_exercise.py localhost 10001
```

Expected Output:

The client terminal will show a series of math problems being sent and their results being received, proving that it successfully found and connected to the TCP server.