**TCP Server-Client Communication:**

**Problem Statement: Write a TCP server and client program in C++ where the server listens for incoming connections and echoes back any message it receives from the client. The client should be able to send a message to the server and display the echoed message.**

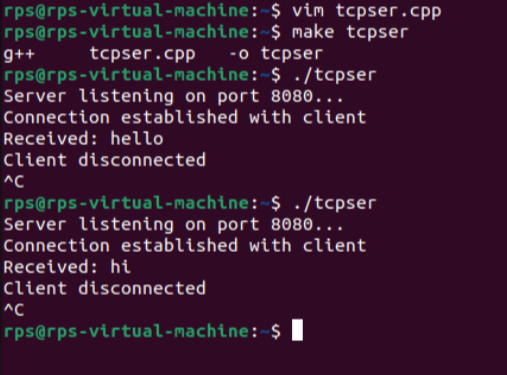
**Requirements:**

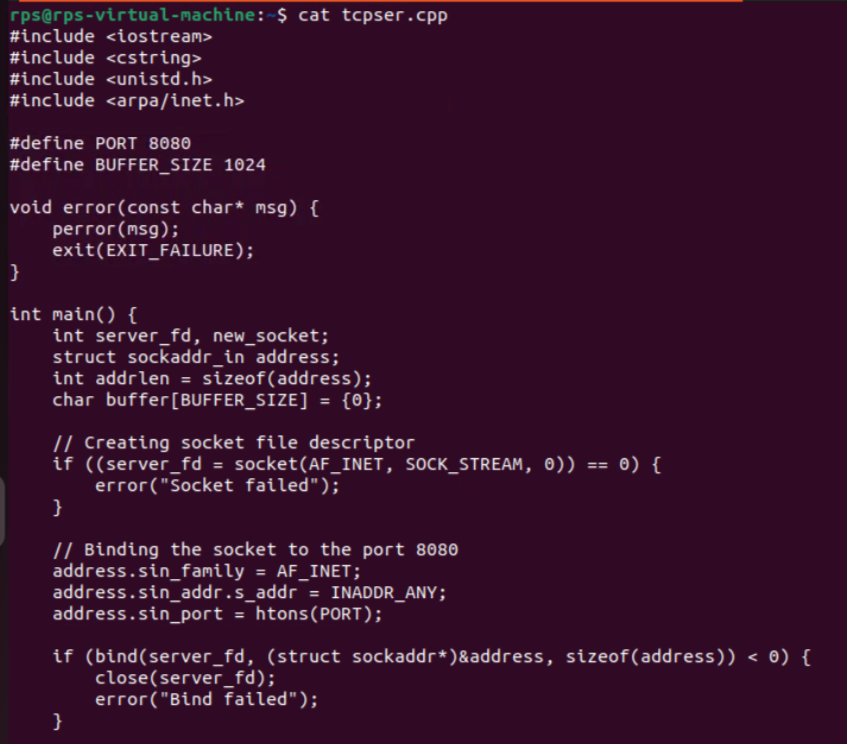
**The server should run indefinitely, waiting for client connections.**

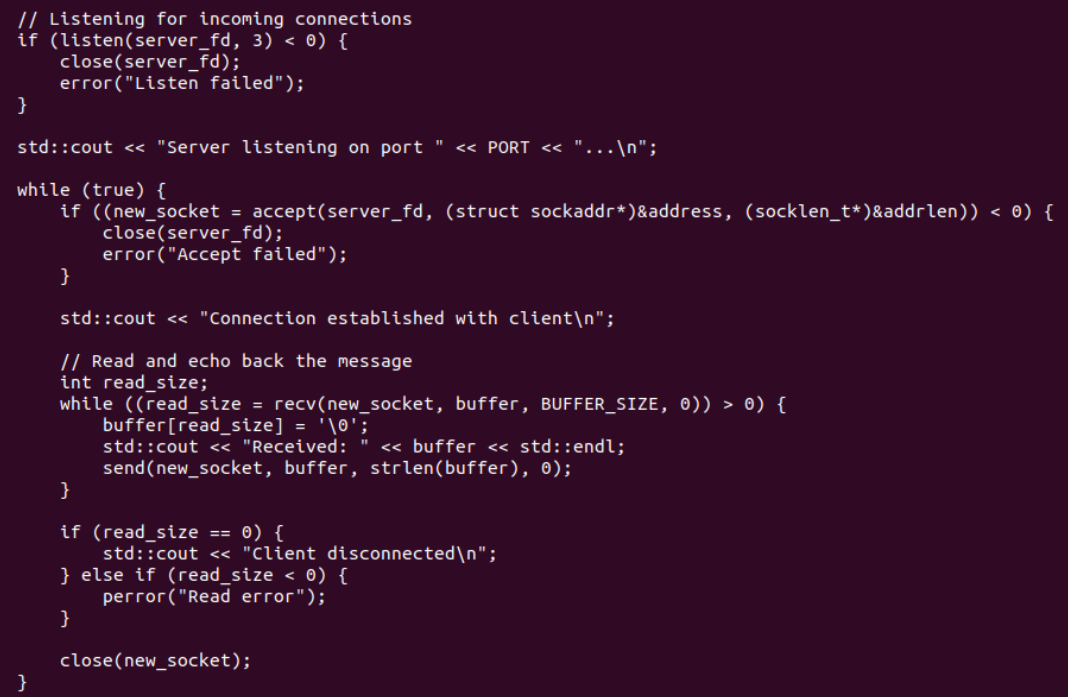
**The client should take a message as input from the user, send it to the server, and display the response.**

**Implement proper error handling and cleanup (e.g., closing sockets).**

**Server:**

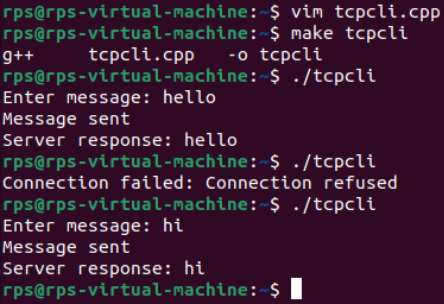
****

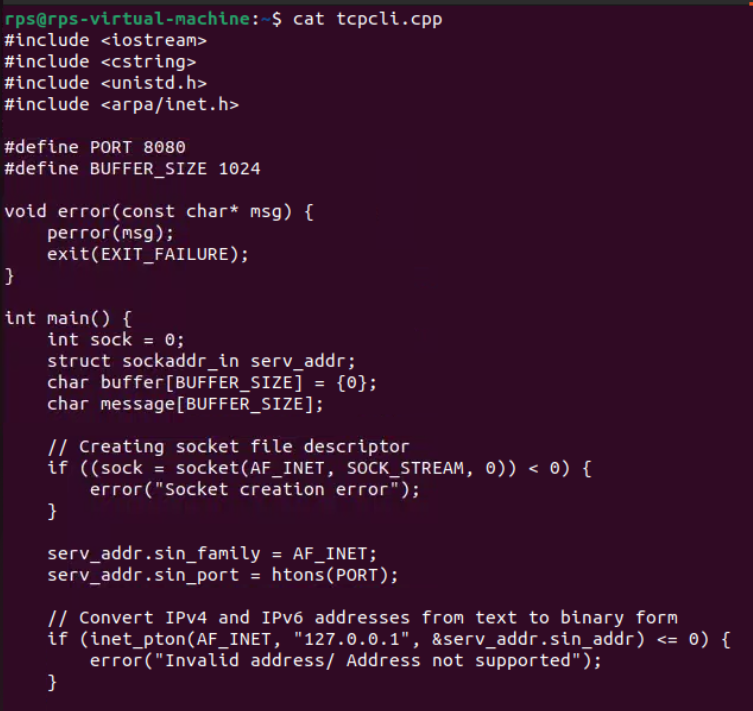
****

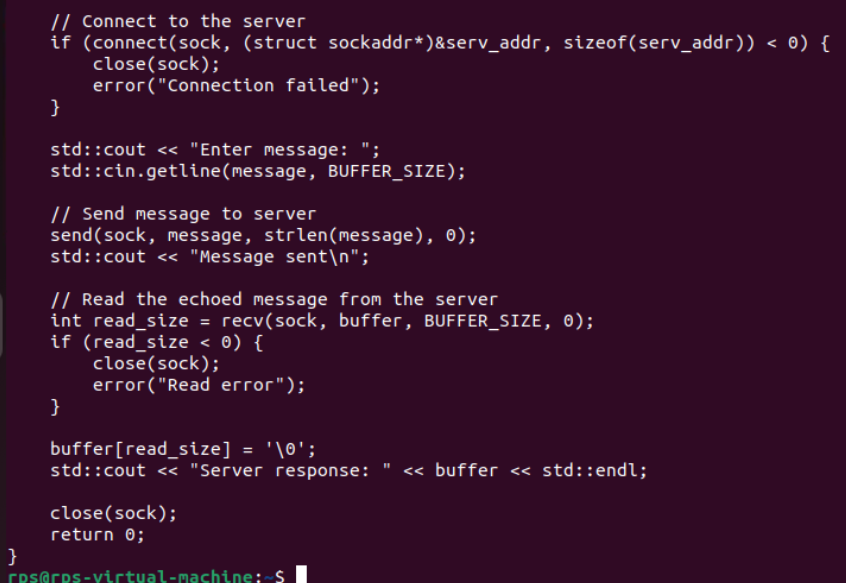
****

****

**Client:**

****

****

****

**UDP Server-Client Communication:**

**Problem Statement: Write a UDP server and client program in C++ where the server listens on a specific port and responds with "Hello, Client!" whenever it receives a message. The client should send a message to the server and print the response.**

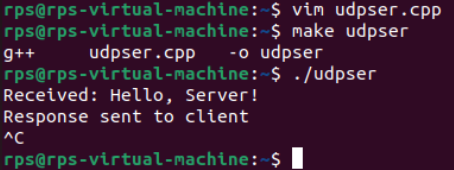
**Requirements:**

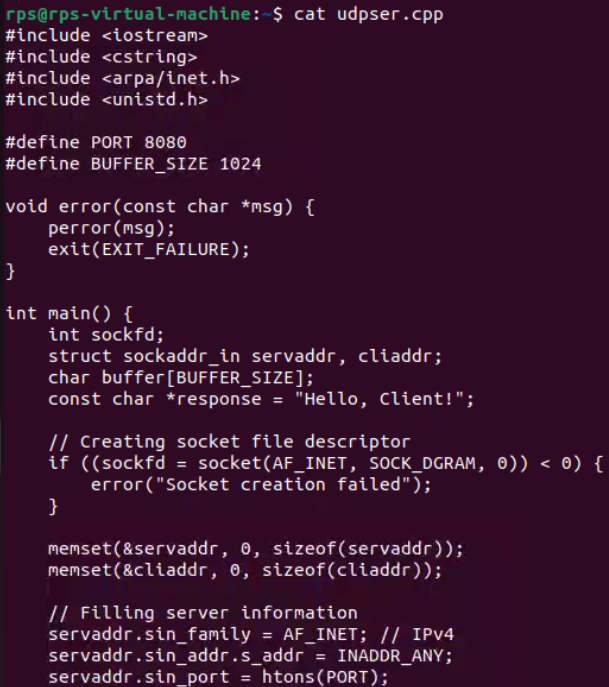
**The server should run indefinitely, waiting for incoming messages.**

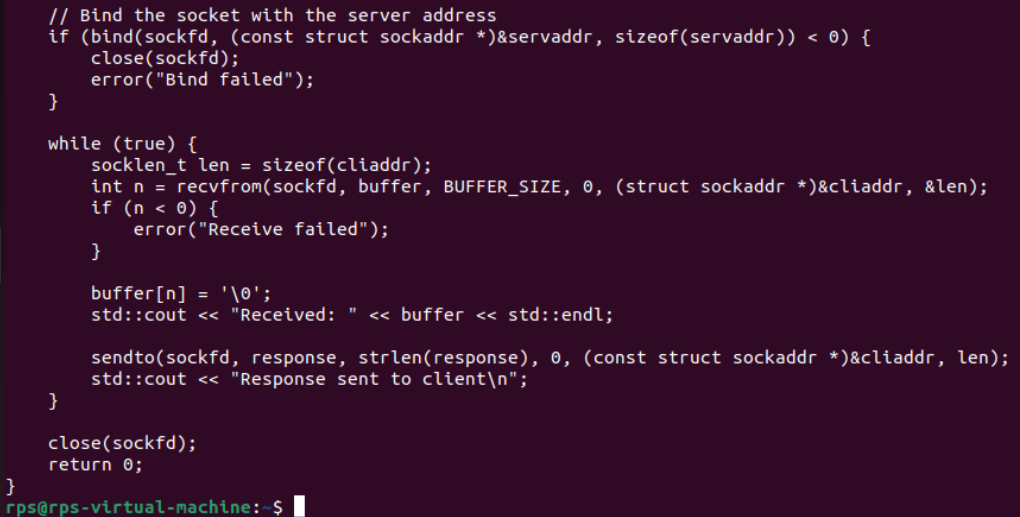
**The client should send a predefined message (e.g., "Hello, Server!") and display the server's response.**

**Implement proper error handling.**

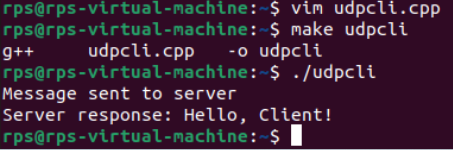
**Server:**

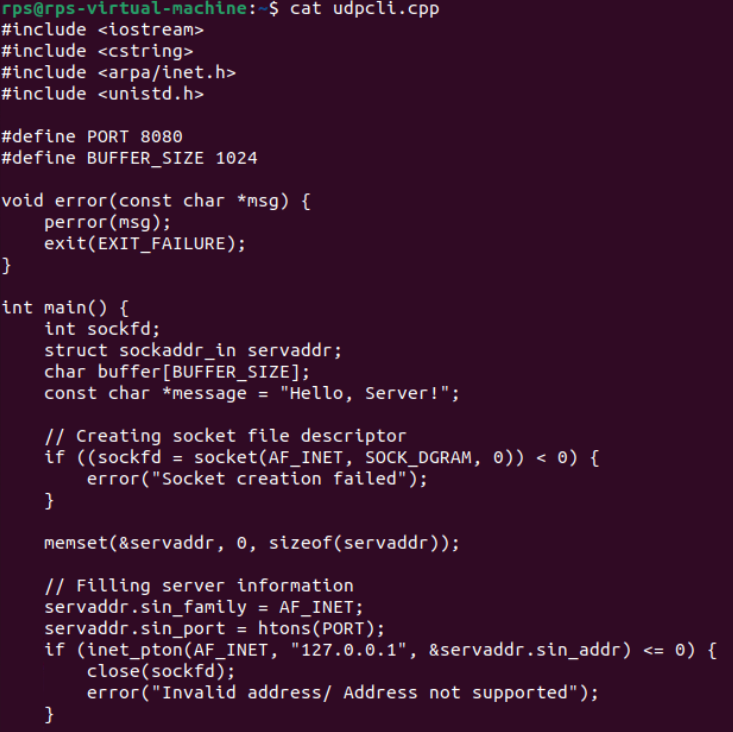
****

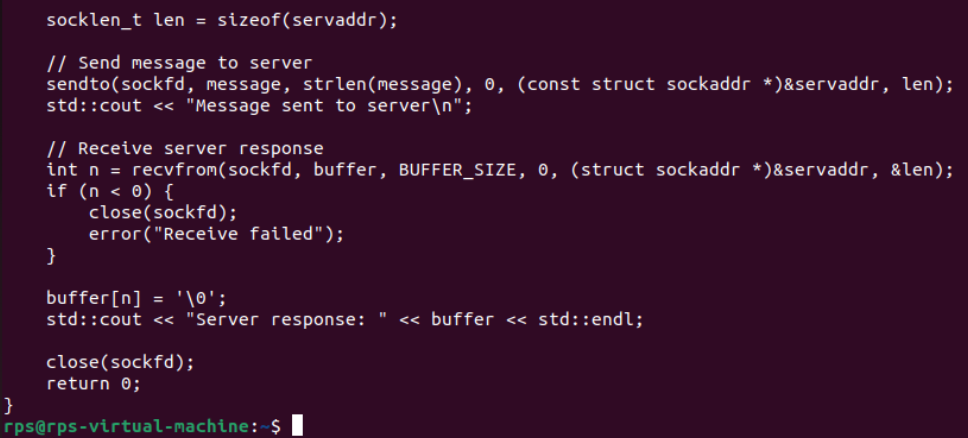
****

****

**Client:**

****

****

****

**File Transfer using TCP:**

**Problem Statement: Write a TCP server and client program in C++ to transfer a file from the client to the server. The server should save the received file with the same name, and the client should specify the file to be sent.**

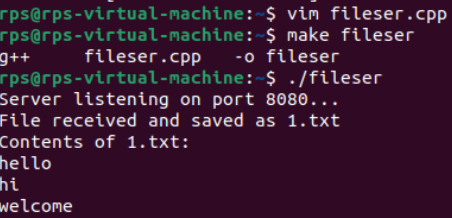
**Requirements:**

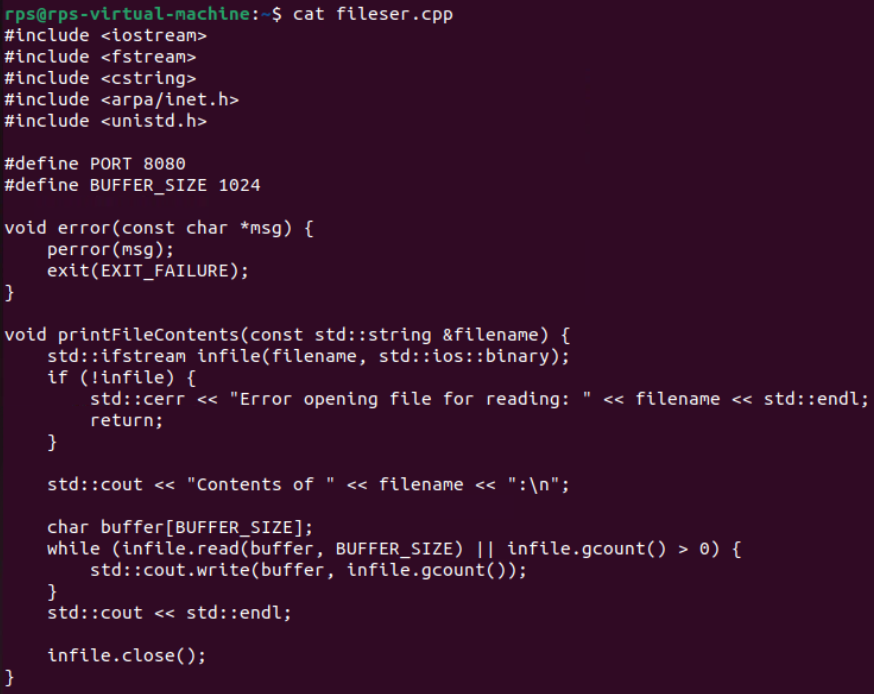
**The server should run indefinitely, waiting for file transfer requests.**

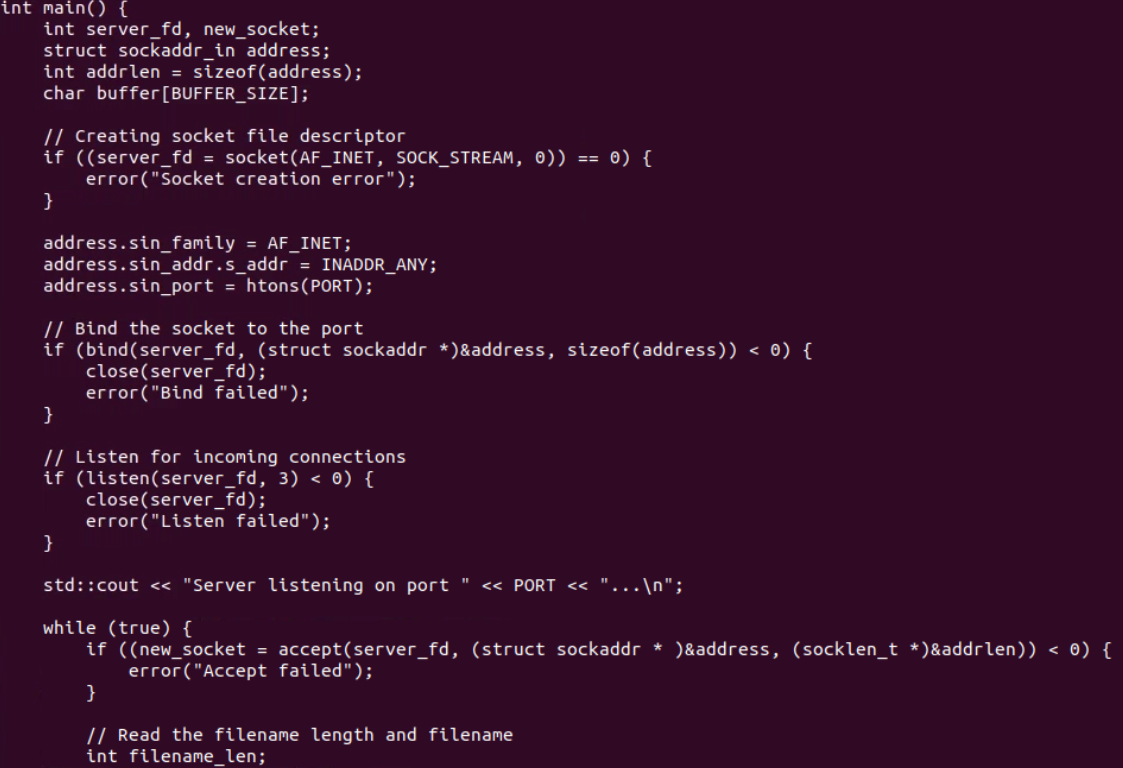
**The client should prompt the user for a file path, read the file, and send its contents to the server.**

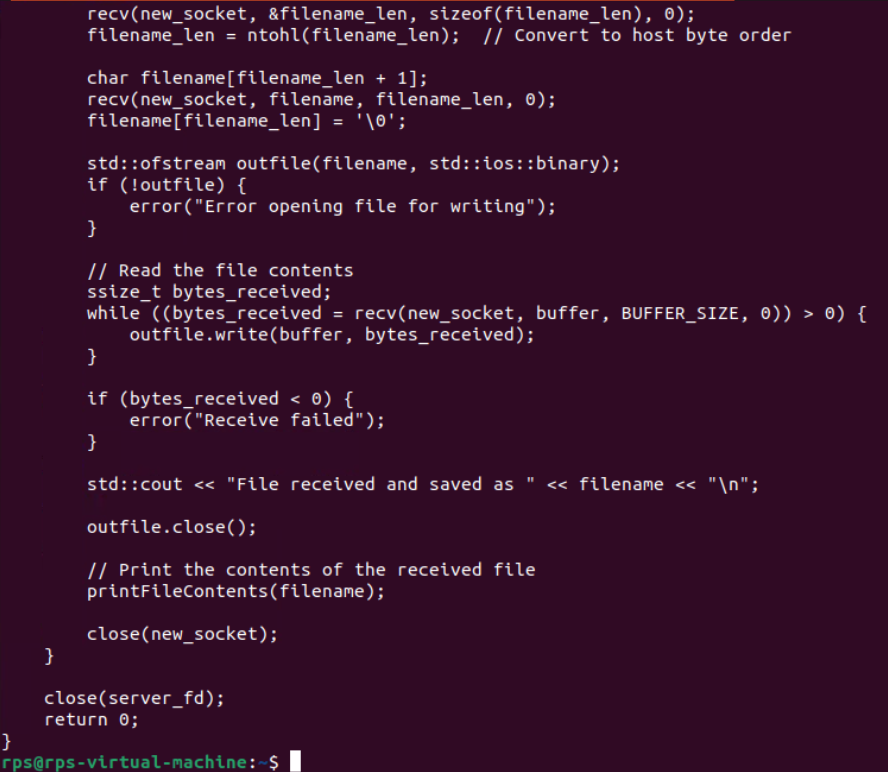
**Implement proper error handling and file operations.**

**Server:**

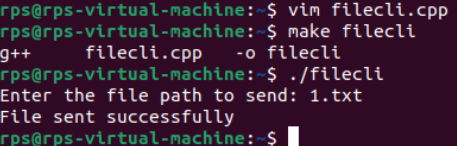
****

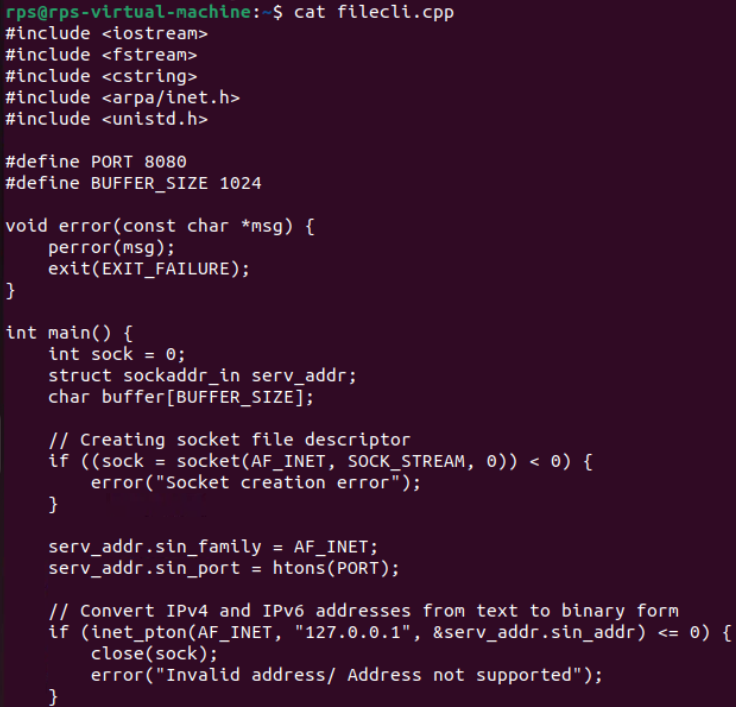
****

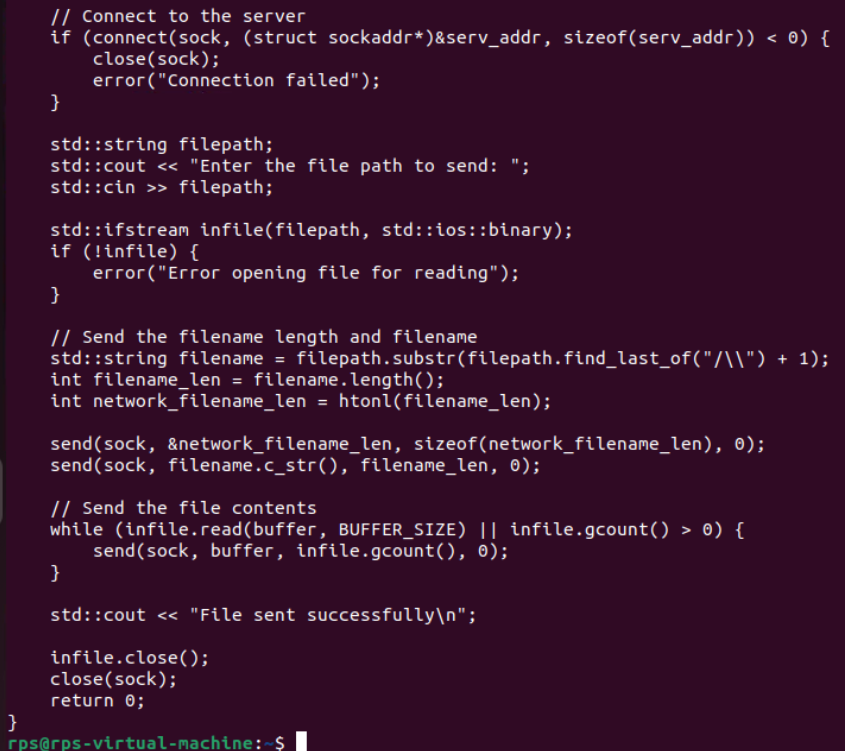
****

****

**Client:**

****

****

****

**Broadcast Messaging using UDP:**

**Problem Statement: Write a UDP server and client program in C to implement a simple broadcast messaging system. The server should broadcast a message to all clients in the network, and each client should display any broadcast messages it receives.**

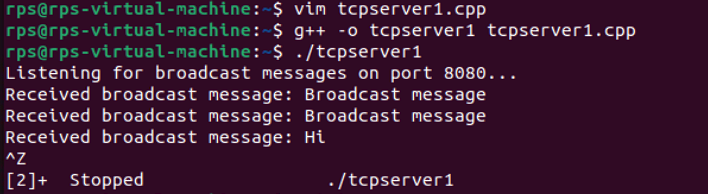
**Requirements:**

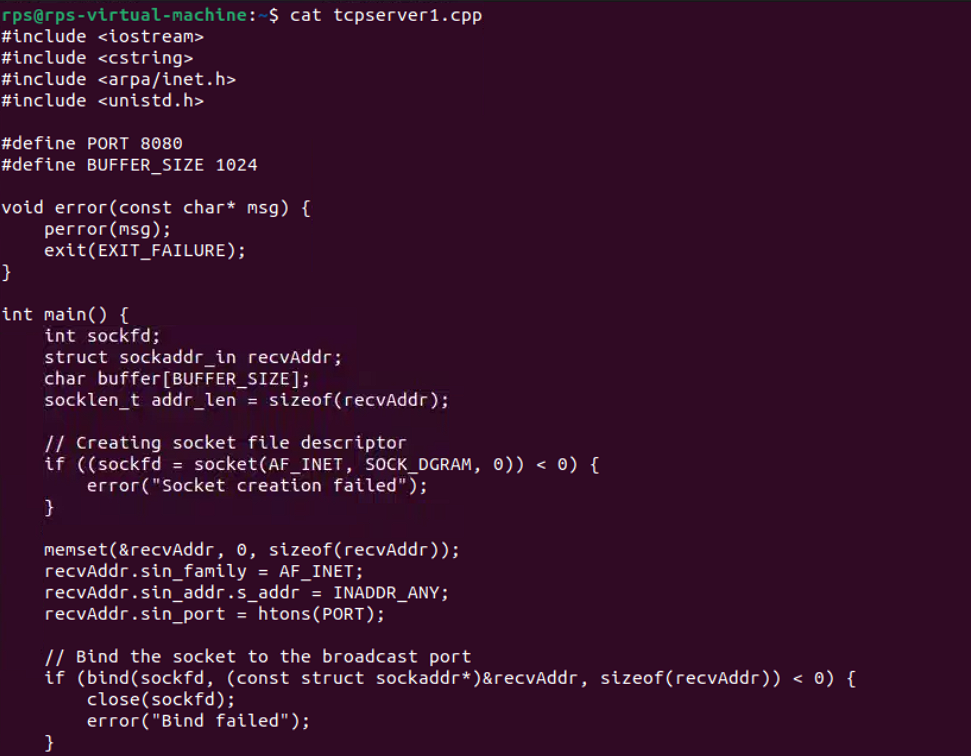
**The server should send a broadcast message to a specific port.**

**Each client should listen on the same port and display any messages it receives.**

**Implement proper error handling and use UDP broadcast mechanisms.**

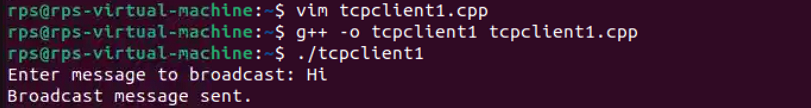
**Server:**

****

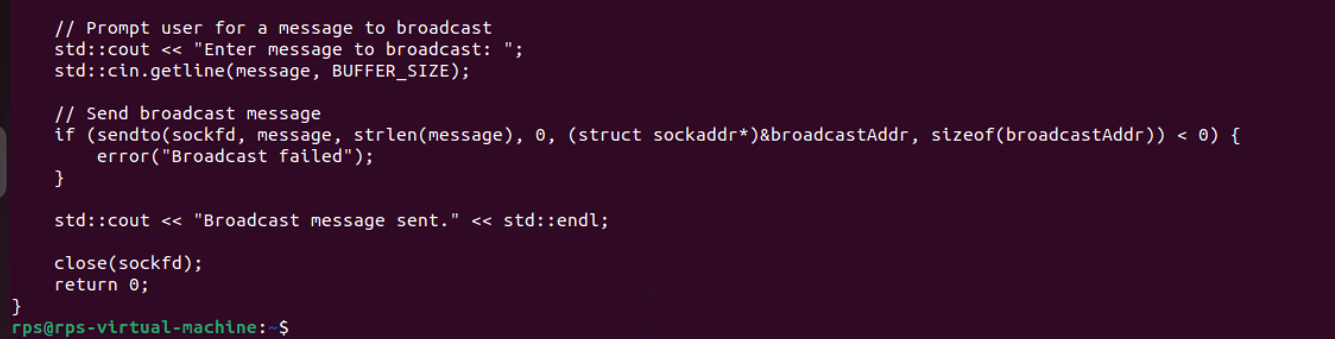
****

****

**Client:**

****

****

****