

```

#include <winsock2.h>
#include <stdio.h>

// Link com a biblioteca Winsock
#pragma comment(lib, "ws2_32.lib")

int main()
{
    WSADATA wsaData;
    SOCKET clientSocket;
    struct sockaddr_in serverAddr;
    char* serverIP = "127.0.0.1"; // Endereço IP do servidor
    int serverPort = 8080;        // Porta do servidor
    char* message = "Olá, servidor!";
    char buffer[1024];
    int recvSize;

    // Inicializa a biblioteca Winsock
    if (WSAStartup(MAKEWORD(2, 2), &wsaData) != 0)
    {
        printf("Falha ao inicializar o Winsock. Código: %d\n", WSAGetLastError());
        return 1;
    }

    // Cria um socket
    clientSocket = socket(AF_INET, SOCK_STREAM, 0);
    if (clientSocket == INVALID_SOCKET)
    {
        printf("Falha ao criar o socket. Código: %d\n", WSAGetLastError());
        WSACleanup();
        return 1;
    }

    // Configura os detalhes do servidor
    serverAddr.sin_family = AF_INET;
    serverAddr.sin_port = htons(serverPort);
    serverAddr.sin_addr.s_addr = inet_addr(serverIP);

    // Conecta ao servidor
    if (connect(clientSocket, (struct sockaddr*)&serverAddr, sizeof(serverAddr)) < 0)
    {
        printf("Falha ao conectar ao servidor. Código: %d\n", WSAGetLastError());
        closesocket(clientSocket);
        WSACleanup();
        return 1;
    }

    printf("Conectado ao servidor %s:%d\n", serverIP, serverPort);

    // Envia uma mensagem ao servidor
    if (send(clientSocket, message, strlen(message), 0) < 0)
    {
        printf("Falha ao enviar dados. Código: %d\n", WSAGetLastError());
        closesocket(clientSocket);
        WSACleanup();
        return 1;
    }

    printf("Mensagem enviada: %s\n", message);

    // Recebe a resposta do servidor (se houver)
    recvSize = recv(clientSocket, buffer, sizeof(buffer) - 1, 0);
    if (recvSize > 0)
    {
        buffer[recvSize] = '\0'; // Garante que a string recebida tenha um terminador nulo
        printf("Resposta recebida: %s\n", buffer);
    }
}

```

```
}  
else if (recvSize == 0)  
{  
    printf("Conexão encerrada pelo servidor.\n");  
}  
else  
{  
    printf("Erro ao receber dados. Código: %d\n", WSAGetLastError());  
}  
  
// Fecha o socket e limpa a biblioteca Winsock  
closesocket(clientSocket);  
WSACleanup();  
  
return 0;  
}
```