```c
//ITERATIVE CLIENT
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <arpa/inet.h>

#define PORT 2000
#define BUFFER_SIZE 1024

int main() {
    int clientSocket;
    char buffer[BUFFER_SIZE];
    struct sockaddr_in serverAddr;
    socklen_t addr_size;

    clientSocket = socket(AF_INET, SOCK_STREAM, 0);
    if (clientSocket < 0) {
        perror("Socket creation failed");
        exit(EXIT_FAILURE);
    }

    serverAddr.sin_family = AF_INET;
    serverAddr.sin_port = htons(PORT);
    serverAddr.sin_addr.s_addr = inet_addr("127.0.0.1");
    memset(serverAddr.sin_zero, '\0', sizeof serverAddr.sin_zero);
    addr_size = sizeof serverAddr;

    if (connect(clientSocket, (struct sockaddr*)&serverAddr, addr_size) < 0) {
        perror("Connection failed");
        close(clientSocket);
        exit(EXIT_FAILURE);
    }

    printf("Enter the message: ");
    fgets(buffer, BUFFER_SIZE, stdin);
    send(clientSocket, buffer, strlen(buffer), 0);
    printf("Message sent to Server\n");

    memset(buffer, 0, BUFFER_SIZE);
    int recv_len = recv(clientSocket, buffer, BUFFER_SIZE, 0);
    if (recv_len > 0) {
        printf("Reply from Server: %s\n", buffer);
    } else {
        perror("Receive failed");
    }

    close(clientSocket);
    return 0;
}
```