Министерство науки и высшего образования Российской Федерации

федеральное государственное автономное   
образовательное учреждение высшего образования   
«Самарский национальный исследовательский университет   
имени академика С.П. Королева»

Институт информатики, математики и электроники

Факультет информатики

Кафедра технической кибернетики

Отчет по лабораторной работе № 4

Дисциплина: «Операционные системы»

Тема «Межпроцессное взаимодействие на основе

каналов и сокетов в операционной системе Windows»

Выполнил: Абрамов Данила Олегович

Группа: 6208-010302D

Самара, 2021

**Задание на лабораторную работу**

**Вариант 12** Сервер должен в режиме эхо-печати (т.е. сразу выводится на экран) получать все введенные данные с клиента, в том числе стирание текста через backspace. Клиент же должен отправлять данные о каждом введенном символе, причём эхопечать должна поддерживать русский язык. Если не поддерживает русский язык, задание может быть зачтено, но по минимальному баллу. Минимум 1 клиент. Если студент хочет сделать много клиентскую версию, то каждый клиент должен выводить свои символы на сервере своим уникальным для него цветом текста.

**Текст программы**

Server

#pragma comment(lib, "ws2\_32.lib")

#pragma warning(disable: 4996)

#include <winsock2.h>

#include <iostream>

using namespace std;

int main() {

// загружаем WSAStartup

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

WSAData wsaData; //создаем структуру для загрузки

WORD word = MAKEWORD(2, 1);

if (WSAStartup(word, &wsaData) != 0) { //проверка на подключение библиотеки

cout << "Error: ошибка подключения библиотеки\n";

WSACleanup();

return 0;

}

SOCKADDR\_IN sockAddr;//получаем информацию об адресе сокета

int sizeOfSockAddr = sizeof(sockAddr); //размер

sockAddr.sin\_addr.s\_addr = inet\_addr("127.1.1.1"); //адрес

sockAddr.sin\_port = htons(1111); //порт

sockAddr.sin\_family = AF\_INET; //семейство протоколов

SOCKET soketListen = socket(AF\_INET, SOCK\_STREAM, 0);//сокет для прослушивания порта

bind(soketListen, (SOCKADDR\*)&sockAddr, sizeof(sockAddr));//функция для привязки сокета

listen(soketListen, SOMAXCONN);//прослушивание, сколько запросов ожидается

SOCKET Connection = accept(soketListen, (SOCKADDR\*)&sockAddr, &sizeOfSockAddr); // подключается клиент

if (Connection == 0) cout << "Error: Ошибка подключения\n"; //не смогли подключиться

else {

cout << "Клиент подключен!\n";

for (int i = 0; i < 15951; ++i) {

char c;

if (recv(Connection, &c, 1, 0) == -1) break;

if (c == '\r') cout << endl;

if (c == '\b') cout << "\b\ ";

cout << c;

}

}

WSACleanup(); //деинициализируем библиотеки

closesocket(Connection); //закрываем сокет

closesocket(soketListen); //закрываем сокет

}

Client

#pragma comment(lib, "ws2\_32.lib")

#pragma warning(disable: 4996)

#include <winsock2.h>

#include <iostream>

#include "conio.h"

using namespace std;

int main() {

// загружаем WSAStartup

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

WSAData wsaData; // создаем структуру для загрузки

WORD word = MAKEWORD(2, 1);

if (WSAStartup(word, &wsaData) != 0) { // проверка на подключение библиотеки

cout << "Error: ошибка подключения библиотеки\n";

WSACleanup();

return 0;

}

SOCKADDR\_IN sockAddr; // получаем информацию об адресе сокета

sockAddr.sin\_addr.s\_addr = inet\_addr("127.1.1.1"); // адрес

sockAddr.sin\_port = htons(1111); // порт

sockAddr.sin\_family = AF\_INET; // семейство протоколов

SOCKET Connection = socket(AF\_INET, SOCK\_STREAM, 0); // создаем сокет для соединения с сервером

if (connect(Connection, (SOCKADDR\*)&sockAddr, sizeof(sockAddr)) != 0) cout << "Error: Ошибка подключения\n";

else {

cout << "Подключение произошло успешно!\n";

char str[15951];

for (int i = 0; i < 15951; ++i) {

char c;

str[i] = getch();

if (str[i] == '\r') cout << endl;

if (str[i] == '\b') cout << "\b\ ";

c = str[i];

send(Connection, &c, 1, 0);

cout << str[i];

}

}

WSACleanup(); // деинициализируем библиотеки

closesocket(Connection); // закрываем сокет

}