ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ ИМ. ПРОФ. М.А. БОНЧ-БРУЕВИЧА»

(СПбГУТ)

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Факультет Информационных технологий и программной инженерии

Кафедра Систем обработки данных

**Дисциплина «Технологии программирования»**

**ОТЧЕТ**

**за практическое занятие №2**

**Тема: Взаимодействие функций в программах**

Выполнил

Студент 2 курса, гр. ИСТ-422

\_\_\_\_\_\_\_\_\_\_\_\_\_ Ващук А.А.

Принял

Доцент кафедры БИС

\_\_\_\_\_\_\_\_\_\_\_\_\_ Медведев В.А.

Санкт-Петербург

2025

**Тема и цель практического занятия:**

Работа с функциями в программе. Получение практических навыков проектирования взаимодействия функций в программах с использованием локальных и глобальных переменных.

**Задание на практическое занятие:**

**Вариант А.**

Каждое из первых двух заданий следует оформить в виде отдельной функции. *В главной функции задавать исходные данные и выводить результаты на экран.*

Программу разработать с функциями в двух версиях в рамках одного приложения. В первой версии глобальные переменные не использовать. Во второй версии взаимодействие функций организовать через глобальные переменные (общую область памяти).

**Задание 1.** Задать данные указанного типа (Приложение А), вычислить их сумму и определить размер отведённой для неё памяти. Вывести на экран значение суммы.

**Задание 2.** Заполнить пятиэлементный массив с клавиатуры и определить *Вычисляемый показатель* (Приложение А).

**Задание 3.** Объединить обе версии в одну программу. Разработать меню для управления программой, в котором по желанию пользователя обеспечить:

1. выполнение любого из заданий для любой из версий и

в любом порядке;

1. выход из программы.

**Текст программы с комментариями:**

#include <iostream>

#include <ctime>

int globalA;

unsigned int globalB;

long double globalMas[5];

// Задание 1

void Input(int &a, unsigned int &b)

{

    std::cout << "Input A and B: ";

    std::cin >> a >> b;

}

void sumV1(int a, unsigned int b)

{

    int result = a + b;

    if (result >= 32 && result <= 126){

        std::cout << "Sum in type char: " << char(result) << "\nSizeof: " << sizeof(result) <<"\n";

    } else std::cout << "Error!\n";

}

void sumV2()

{

    char result = globalA + globalB;

    if (result >= 32 && result <= 126){

        std::cout << "Sum in type char: " << char(result) << "\nSizeof: " << sizeof(result) << "\n";

    } else std::cout << "Error!\n";

}

// Задание 2

long double maxV1(long double (&mas)[5])

{

    std::cout << "Enter 5 values: ";

    for (int i = 0; i < 5; i++){

        std::cin >> mas[i];

    }

    long double max = mas[0];

    for (int i = 1; i < 5; i++){

        if (max < mas[i]) max = mas[i];

    }

    return max;

}

long double maxV2()

{

    std::cout << "Enter 5 values: ";

    for (int i = 0; i < 5; i++){

        std::cin >> globalMas[i];

    }

    long double max = globalMas[0];;

    for (int i = 1; i < 5; i++){

        if (max < globalMas[i]) max = globalMas[i];

    }

    return max;

}

// int в char

void Task5(int &button5, char &button)

{

    switch (button5)

    {

    case 1: button = '1'; break;

    case 2: button = '2'; break;

    case 3: button = '3'; break;

    case 4: button = '4'; break;

    default: break;

    }

}

// Задание 3

void menu(int &a, unsigned int &b, long double (&mas)[5])

{

    // Переменные, которые нужны для работы меню

    char button;

    int button5;

    bool flag5 = true;

    // Само меню вызывается в цикле

    while (true){

        // Проверка на рандом в меню

        if (flag5){

            std::cout << "\n1. Task 1 (V1)\n"<< "2. Task 1 (V2)\n" << "3. Task 2 (V1)\n" << "4. Task 2 (V2)\n" << "5. Random\n" << "Q. Exit\n";

            std::cin >> button;

        } else{

            flag5 = true;

            Task5(button5, button);

        }

        if (button == 'Q') break;

        system("cls");

        switch (button)

        {

        case '1':

            Input(a, b);

            sumV1(a, b);

            break;

        case '2':

            Input(globalA, globalB);

            sumV2();

            break;

        case '3':

            std::cout << maxV1(mas) << "\n";

            break;

        case '4':

            std::cout << maxV2() << "\n";

            break;

        case '5':

            button5 = rand() % 4 + 1;

            flag5 = false;

            break;

        default:

            std::cout << "Error!\n";

        }

    }

}

int main()

{

    // Создание переменных

    int a;

    unsigned int b;

    long double mas[5];

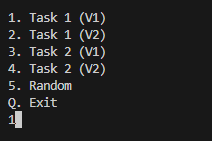
    // Вызов меню

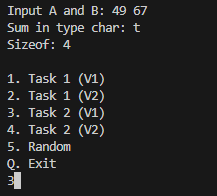
    menu(a, b, mas);

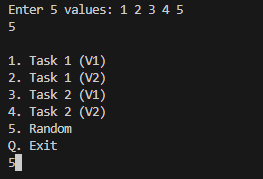
    return 0;

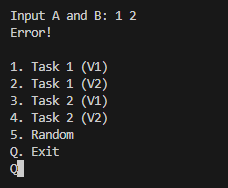
}

**Результаты работы:**









**Выводы по созданному проекту и использованию средств языка программирования:**

Я использовал функции, в которых реализовал разные задачи с локальными и глобальными переменными. Глобальные переменные удобно использовать, но как по мне не очень практично из-за выделения памяти при ее объявлении. Можно использовать ссылки, чтобы выделять память при создании объекта.