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 №3**

**Тема: Ввод-вывод и форматирование данных**

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:**

Получение практических навыков работы с файлами и использования средств форматирования данных.

**Задание на практическое занятие:**

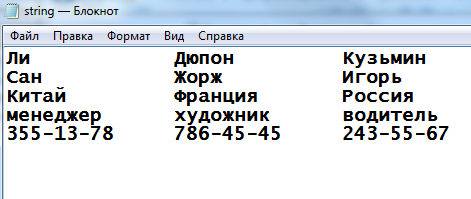
*Каждый пункт задания выполнить в виде отдельной функции.*

**Вариант А.**

**Задание 1.** Согласно индивидуальному номеру (см. таблица 4), используя *манипуляторы ввода-вывода, в цикле* сформировать матрицу на экране (без пробелов между элементами).

**Задание 2.** Разработать функцию преобразования *исходной строки* (согласно индивидуальному номеру, см. таблица 5), используя *только* библиотечные функции (обращение к элементам символьного массива по индексу запрещено). При этом для исходной строки *динамически выделить память в главной функции*.

**Задание 3.** Спроектировать структуру, описывающую характеристики «*Предмета»* (см. таблица 5). Создать массив из 5-ти структурных переменных и записать их в файл, как показано на рисунке ниже. Определить *Вычисляемый показатель*.



**Задание 4.** Разработать меню, в котором по желанию пользователя:

1. выполнить любой пункт задания;
2. прочитать из файла информацию о любом из «предметов»;

выйти из программы.

**Текст программы:**

#include <iostream>

#include <cstring>

#include <fstream>

int num;

struct Zadanie3

{

    char nameDoc[10];

    int year;

};

struct Zadanie3 init()

{

    struct Zadanie3 temp;

    std::cout << "Enter name doc and year: ";

    std::cin >> temp.nameDoc;

    std::cin >> temp.year;

    return temp;

};

void Parsing()

{

    std::ifstream from2("string.txt");

    if (!from2) {

        std::cout << "Error of file opening" << std::endl;

        return;

    }

    else

    {

        char names[5][20];

    for (int i = 0; i < 5; i++) {

        from2 >> names[i];

    }

    int years[5];

    for (int i = 0; i < 5; i++) {

        from2 >> years[i];

    }

    for (int i = 0; i < 5; i++) {

        std::cout << i + 1 << ". " << names[i] << "\n";

    }

    std::cout << "Enter a number (1-5): ";

    std::cin >> num;

    if (num < 1 || num > 5) {

        std::cout << "Invalid number!" << std::endl;

        return;

    }

    system("cls");

    std::cout << "File:\n";

    std::cout << names[num - 1] << " " << years[num - 1] << std::endl;

    std::cout << "\n";

    from2.close();

    }

}

void FileAndMas(Zadanie3 mas3[5])

{

    std::ofstream in("string.txt");

    if (!in) {

        std::cout << "Error of file opening" << std::endl;

        return;

    }

    else

    {

        for (int i = 0; i < 5; i++) {

            in << mas3[i].nameDoc << "\t";

        }

        in << "\n";

        for (int i = 0; i < 5; i++) {

            in << mas3[i].year << "\t";

        }

        in << "\n";

        in.close();

    }

    std::ifstream from("string.txt");

    if (!from) {

        std::cout << "Error of file opening" << std::endl;

        return;

    }

    else

    {

        char names[5][20];

        for (int i = 0; i < 5; i++) {

            from >> names[i];

        }

        int years[5];

        for (int i = 0; i < 5; i++) {

            from >> years[i];

        }

        std::cout << "Documents 2024:\n";

        for (int i = 0; i < 5; i++) {

            if (years[i] == 2024) {

                std::cout << names[i] << " " << years[i] << std::endl;

            }

        }

        std::cout << "\n";

        from.close();

    }

}

void matrix()

{

    char M[6][6];

    int n = 1;

    for (int i = 0; i < 6; i++)

    {

        M[i][0] = '0' + (6 - i);

    }

    for (int i = 0; i < 6; i++)

    {

        for (int j = 1; j < (6 - i); j++)

        {

            M[i][j] = '?';

        }

    }

    for (int i = 0; i < 6; i++)

    {

        for (int j = 0; j < 6 - i; j++)

        {

            std::cout << M[i][j];

        }

        std::cout << std::endl;

    }

}

void Stroka(char \*stroka)

{

    char \*zeroStart = strstr(stroka, "Zero");

    char reversedZero[5] = {0};

    char \*p = zeroStart + 3;

    char \*q = reversedZero;

    while (p >= zeroStart)

    {

        \*q = \*p;

        q++;

        p--;

    }

    \*q = '\0';

    strncpy(zeroStart, reversedZero, 4);

    std::cout << "Changed line: " << stroka << std::endl;

}

int main()

{

    char button;

    Zadanie3 mas3[5];

    bool isMas3Initialized = false;

    char \*stroka = new char[20];

    strcpy(stroka, "Zero Five Null");

    std::cout << "1. Matrix\n2. String conversions\n3. File\nb. Parsing\nC. Exit\nInput: ";

    std::cin >> button;

    while (button != 'C')

    {

        system("cls");

        switch (button)

        {

        case '1':

            matrix();

            std::cout << "\n";

            break;

        case '2':

            Stroka(stroka);

            std::cout << "\n";

            break;

        case '3':

            if (!isMas3Initialized) {

                for (int i = 0; i < 5; i++) {

                    mas3[i] = init();

                }

                isMas3Initialized = true;

            }

            system("cls");

            FileAndMas(mas3);

            break;

        case 'b':

            if (!isMas3Initialized) {

                std::cout << "Array is not initialized. Please use option 3 first.\n";

            } else {

                Parsing();

            }

            break;

        default:

            std::cout << "Invalid operator" << std::endl;

            break;

        }

        std::cout << "1. Matrix\n2. String conversions\n3. File\nb. Parsing\nC. Exit\nInput: ";

        std::cin >> button;

    }

    delete[] stroka;

    return 0;

}

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

В программе были использованы такие технологии, как: матрица, работа с библиотекой <cstring>, структура данных, работа с файлом.

Была создана матрица 6x6, заполненная некоторыми элементами. Вывод осуществляется благодаря «манипуляторам ввода-вывода». Такие как: setw, setfill, endl.

Была динамически выделена память для исходной строки. Были использованы: strncpy, strstr.

В программе реализована структура «Предмета», в которой описано название и дата документа. Для хранения использовался массив размером (5). Далее в программе происходит запись и считывание в файл.

В программе разработано меню, в котором есть все 3 пункта: a, b, c.