



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н. Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н. Э. Баумана)

ФАКУЛЬТЕТ «Информатика, искусственный интеллект и системы управления» (ИУ)
КАФЕДРА «Информационная безопасность» (ИУ8)

Лабораторная работа №1
ПО КУРСУ
«Алгоритмические языки»
на тему «Изучение классов языка Си++»

Студент

ИУ8-21
(Группа)

Г. А. Карев
(И. О. Фамилия)

Преподаватель:

В. В. Соборова
(И.О. Фамилия)

Цель работы:

Цель работы состоит в изучении основных понятий ООП языка Си++ – классов и объектов, и овладении навыками разработки программ на языке Си++ с использованием объектно-ориентированных средств.

Задача:

Вариант 11

Класс – межсетевой экран (МЭ). Параметры (поля класса) – название и номер класс защищенности (Существует пять классов защищенности МЭ, наивысший 1-ый, самый низкий 5, например, если требуется использовать МЭ 3- го класса защищенности, то можно использовать МЭ с классами 1, 2 или 3). Статус доступа всех полей private. Класс включает: конструктор, при необходимости функции доступа к полям, функцию, проверяющую можно ли МЭ использовать для заданного класса (номер заданного класса – параметр функции), функцию печати параметров МЭ. Вывести на печать параметры тех МЭ, которые можно использовать для заданного класса защищенности, номер класса защищенности вводится с клавиатуры.

Код:

```
#include <iostream>
#include <vector>
#include <Windows.h>
#include <string>
#include <fstream>
using namespace std;

class firewall {
    string name;
    int type;

public:
    firewall() {
        name = "";
        type = 0;
    }
    void set(string na, int ty) {
        name = na;
        type = ty;
    }

    firewall(string na, int ty) {
        name = na;
        type = ty;
    }

    void print() const {
        cout << name << " - " << "Класс защиты: " << type << endl;
    }
    void fprint(ofstream& out) const {
        out << name << " - " << "Класс защиты: " << type << endl;
    }
}
```

```

        bool check(int t) {
            if (type > t) { return true; }
            else { return false; }
        }
};

int main(int argc, const char* argv[]) {
    SetConsoleOutputCP(CP_UTF8);
    SetConsoleCP(CP_UTF8);

    if (argc != 3) { cerr << "Инвадид количество параметров" << endl; }
    const string fileCH(argv[1]);
    const string fileZ(argv[2]);
    int t;
    cout << "Введите необходимый класс защиты: "; cin >> t;

    string name1;
    int type1;
    firewall* pfirewall;

    ifstream fin(fileCH);
    if (!fin) {
        cerr << "Ошибка доступа к файлу.";
        system("pause");
        return 1;
    }

    ofstream fout(fileZ);
    if (!fout) {
        cerr << "Ошибка доступа к файлу.";
        system("pause");
        return 1;
    }

    int n;
    fin >> n;
    pfirewall = new firewall[n];
    vector <firewall> firewallData(n);
    for (int i = 0; i < n; i++) {
        fin >> name1;
        fin >> type1;
        pfirewall[i].set(name1, type1);
        firewallData[i].set(name1, type1);
    }

    for (int i = 0; i < n; i++) {
        if (firewallData[i].check(t) == true) {
            cout << "\nFirewal #" << i + 1 << endl;
            firewallData[i].print();
            fout << "\nFirewall #" << i + 1 << endl;
            firewallData[i].fprint(fout);
        }
    }
    return 0;
};

```

Результаты вывода программы:

```
5
Базальт 1
Гранит 2
Мрамор 3
Золото 4
Алмаз 5
```

- данные из списка МЭ

Ввод в консоль.

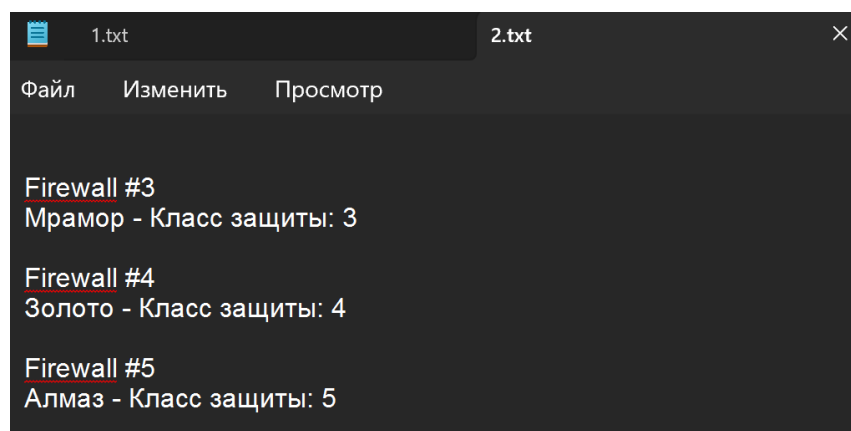
```
PS C:\Users\Admin\source\repos\ammo_VS\ammo_VS> ./a.exe 1.txt 2.txt
Введите необходимый класс защиты: 2

Firewal #3
Мрамор - Класс защиты: 3

Firewal #4
Золото - Класс защиты: 4

Firewal #5
Алмаз - Класс защиты: 5
PS C:\Users\Admin\source\repos\ammo_VS\ammo_VS> |
```

Выведенные в текстовый файл подходящие данные.



Вывод:

Научился разрабатывать программы на языке Си, использующие основные средства ООП. Использовал на практике конструктор и чтение/запись из/в файл.