

# Лабораторная работа 6

Выполнил Семенов М.С. СКБ223

## 1) Использование программы

Флаги:

- MIEM — сгенерировать номер студака МИЭМ
- MGUU — сгенерировать номер студака МГТУ
- tofile — записать результат в файл (будет сохранен в файл «result.txt»)
- fromfile + <Имя файла> — считать данные из файла

Пример:

```
./main --MIEM -- fromfile «test.txt»
```

## 2) Логика программы

От виртуального класса Studak наследуются классы MIEM и MGUU, которые переопределяют методы pseudo\_generator() и print(). Первый генерирует псевдослучайное число на основе даты рождения, второй число С и возвращает результат. Метод pseudo\_generator сначала генерирует случайное восьмизначное число, потом при помощи побитовой операции XOR генерирует псевдослучайное число и возвращает его. Методы sex\_converter и date\_converter приводят данные к удобному виду.

## 3) Код

```
#include <iostream>
#include <random>
#include <ctime>
#include <string>
#include <cstring>
#include <fstream>

#define flag1 "--MIEM"
#define flag2 "--MGUU"
#define flag3 "--tofile"
#define flag4 "--fromfile"

using namespace std;

class Studak{
public:
    virtual string pseudo_generator() = 0;
    virtual string print() = 0;
    int date_converter(int YYYY, int MM, int DD){
        return YYYY*10000 + MM*100 + DD;
    }
    int sex_converter(string sex, int man, int woman){
        if (sex == "man"){
            return man;
        }
        else {
            return woman;
        }
    }
}
```

```
};
```

```
class MIEM: public Studak{
private:
    string sex;
    int DD, MM, YYYY, C, man = 8, woman = 4;

public:
    MIEM(string user_sex, int user_YYYY, int user_MM, int user_DD):
        sex(user_sex),
        DD(user_DD),
        MM(user_MM),
        YYYY(user_YYYY){};

    string pseudo_generator() override {
        srand(time(0));
        int temp = rand()%999999999+100000000;
        string temp2 = to_string(date_converter(YYYY, MM, DD) ^ temp);
        string result = temp2.substr(temp2.length()-5);
        return result;
    }

    string print() override {
        string temp, temp3, result;
        int temp2 = 0;
        temp += to_string(sex_converter(sex, man, woman));
        temp += to_string(date_converter(YYYY, MM, DD));
        temp += pseudo_generator();
        for (int i = 0; i < temp.length(); i++){
            temp2 += (temp[i] - '0')*(i+1);
        }
        for (int i = 0; i < 10; i++){
            if ((temp2 + i*15)%11 == 0){
                temp3 = to_string(i);
            }
        }
        result = temp + temp3;
        return result;
    }
};
```

```
class MGTUU: public Studak{
private:
    string sex;
    int DD, MM, YYYY, C, man = 2, woman = 1;

public:
    MGTUU(string user_sex, int user_YYYY, int user_MM, int user_DD):
        sex(user_sex),
        DD(user_DD),
        MM(user_MM),
        YYYY(user_YYYY){};

    string pseudo_generator() override {
        srand(time(0));
        int temp = rand()%999999999+100000000;
        string temp2 = to_string(date_converter(YYYY, MM, DD) ^ temp);
        cout << temp2 << endl;
        string result = temp2.substr(temp2.length()-4);
        return result;
    }
};
```

```

    }

    string print() override {
        string temp, temp3, result;
        int temp2 = 0;
        temp += to_string(sex_converter(sex, man, woman));
        temp += to_string(date_converter(YYYY, MM, DD));
        temp += pseudo_generator();
        for (int i = 0; i < temp.length(); i++){
            temp2 += (temp[i] - '0')*(i+1);
        }
        for (int i = 0; i < 10; i++){
            if ((temp2 + i*14)%10 == 0){
                temp3 = to_string(i);
                break;
            }
            else{
                temp3 = '?';
            }
        }
        result = temp + temp3;
        return result;
    }
};

```

```

int main(int argc, char* argv[]){
    if (argc < 3 || argc > 5){
        cout << "Wrong input" << endl;
    }
    else {
        if(argc == 3 && strcmp(argv[2], flag3)==0){
            ofstream file("result.txt");
            if (file.is_open()){
                string sex;
                int YYYY, MM, DD;
                cout << "Enter sex and birthday: " << endl;
                cin >> sex >> YYYY >> MM >> DD;
                if(strcmp(argv[1], flag1)==0){
                    MIEM studak(sex, YYYY, MM, DD);
                    file << studak.print();
                }
                else {
                    MGTUU studak(sex, YYYY, MM, DD);
                    file << studak.print();
                }
            }
            else {
                cout << "Error while opening file" << endl;
            }
        }
        else if (argc == 4 && strcmp(argv[2], flag4)==0){
            ifstream file(argv[3]);
            if (file.is_open()){
                string sex;
                int YYYY, MM, DD;
                file >> sex >> YYYY >> MM >> DD;
                if(strcmp(argv[1], flag1)==0){
                    MIEM studak(sex, YYYY, MM, DD);
                    cout << studak.print() << endl;
                }
            }
        }
    }
}

```

```

        else {
            MGTUU studak(sex, YYYY, MM, DD);
        }
    }
    else {
        cout << "Error while opening file" << endl;
    }
}
else if(strcmp(argv[4], flag3)==0 && strcmp(argv[2], flag4)==0){
    ifstream file(argv[3]);
    ofstream file1("result.txt");
    if (file.is_open() && file1.is_open()){
        string sex;
        int YYYY, MM, DD;
        file >> sex >> YYYY >> MM >> DD;
        if(strcmp(argv[1], flag1)==0){
            MIEM studak(sex, YYYY, MM, DD);
            file1 << studak.print();
        }
        else {
            MGTUU studak(sex, YYYY, MM, DD);
            file1 << studak.print();
        }
    }
}
else {
    cout << "Wrong input!" << endl;
}
}
return 0;
}

```