

Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ»
Имени Гумарбека Даукеева
Кафедра инженерной кибернетики



AUES
Since 1975

Лабораторная работа №7
Файлы и работа с ними.

Специальность: Автоматизация и управление

Выполнил(-а): Суворов Роман

Группа: АУ-18-5

Вариант: 9

Принял(-а): Рудакова Л.Н.

«____» _____ 20г.

(оценка)

(подпись)

Алматы, 2020г.

7 Лабораторная работа. Файлы и работа с ними

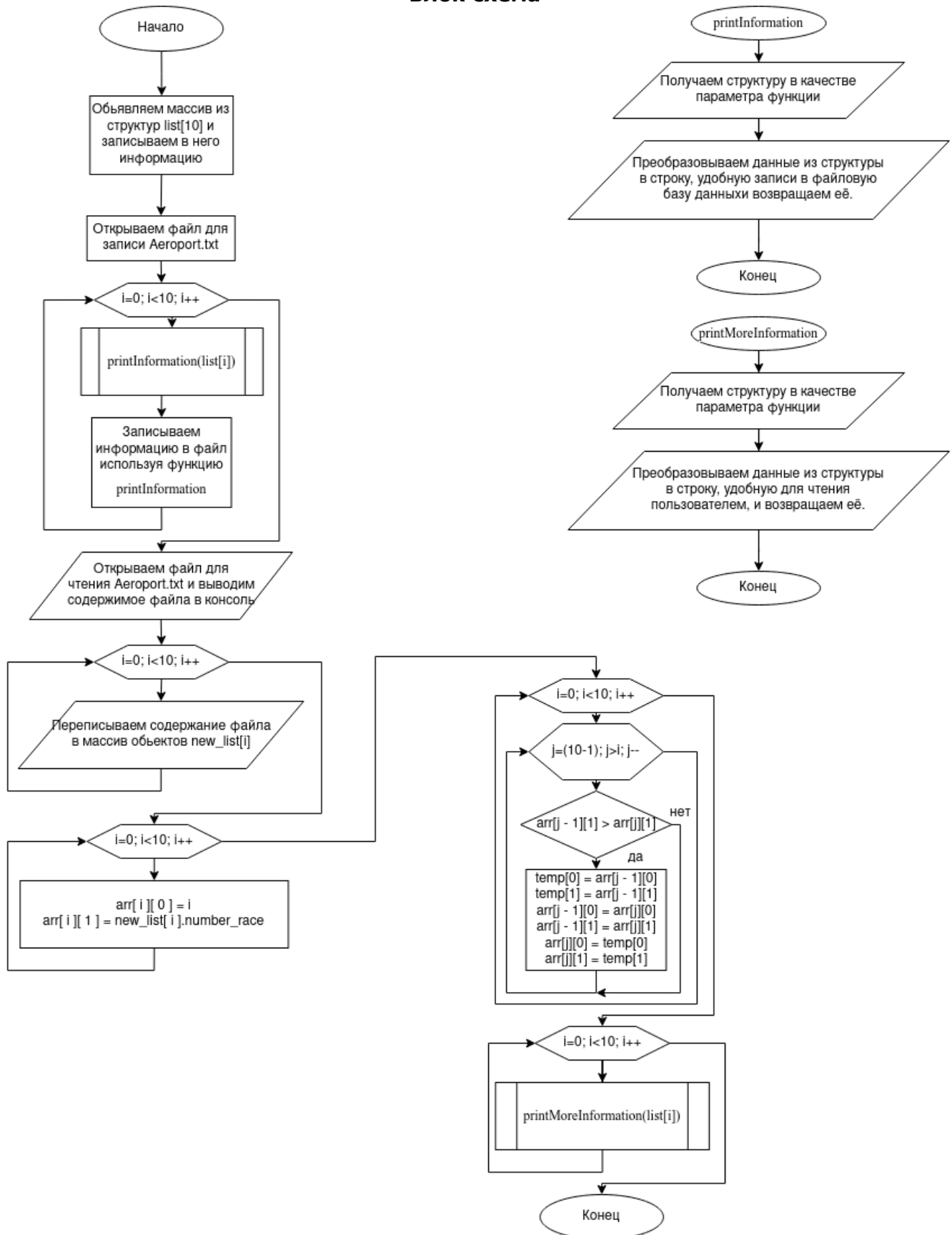
Цель: получить практические навыки работы с файлами в C++.

Задание:

Дан файл, в котором хранится расписание полетов: номер рейса, тип самолета, дни и время вылета рейса, название пунктов вылета-назначения рейса, время в пути. Вывести на экран расписание, отсортированное по номерам рейсов.

Выполнение задания:

Блок схема



Код

```
#include<iostream>
#include<fstream>
#include<string>
using namespace std;

struct Item
{
    string number_race;
    string type;
    string dni_vileta;
    string vremya_vileta;
    string punkt_vileta;
    string naznachenije;
    string vremya_puty;
};

string printInformation(Item item)
{
    return item.number_race + "\n" +
        item.type + "\n" +
        item.dni_vileta + "\n" +
        item.vremya_vileta + "\n" +
        item.punkt_vileta + "\n" +
        item.naznachenije + "\n" +
        item.vremya_puty + "\n";
}

string printMoreInformation(Item item)
{
    return "Nomer reisa: " + item.number_race + "\n" +
        "Tip samolyeta: " + item.type + "\n" +
        "Den' vileta: " + item.dni_vileta + "\n" +
        "Vremya vileta: " + item.vremya_vileta + "\n" +
        "Punkt vileta: " + item.punkt_vileta + "\n" +
        "Punkt naznacheniya: " + item.naznachenije + "\n" +
        "Vremya v puti:" + item.vremya_puty + "\n" +
        "*****" + "\n";
}

int main() {
    struct Item list[10]=
    {
        { "00000002","type01","01.01.01","01.01","punct01","naznachenye01","01h" },
        { "00000001","type02","01.01.02","01.02","punct02","naznachenye02","02h" },
        { "00000003","type03","01.01.03","01.03","punct03","naznachenye03","03h" },
        { "00000007","type04","01.01.04","01.04","punct04","naznachenye04","04h" },
        { "00000005","type05","01.01.05","01.05","punct05","naznachenye05","05h" },
        { "00000006","type06","01.01.06","01.06","punct06","naznachenye06","06h" },
        { "00000004","type07","01.01.07","01.07","punct07","naznachenye07","07h" },
        { "00000008","type08","01.01.08","01.08","punct08","naznachenye08","08h" },
        { "00000009","type09","01.01.09","01.09","punct09","naznachenye09","09h" },
        { "00000010","type10","01.01.10","01.10","punct10","naznachenye10","10h" },
    };
};
```

```

ofstream workf("Aeroport.txt");
for(int i=0; i<10; i++){
    workf << printInformation(list[i]);
}
workf.close();

struct Item new_list[10];

ifstream out("Aeroport.txt");
if (out.is_open())
{
    for(int i=0; i<10; i++){
        out >> new_list[i].number_race >> new_list[i].type >> new_list[i].dni_vileta
>> new_list[i].vremya_vileta >> new_list[i].punkt_vileta >> new_list[i].naznachenije >>
new_list[i].vremya_puty;
    }
}

out.close();

int arr[10][2];

for(int i=0; i<10; i++){
    arr[i][0] = i;
    arr[i][1] = stoi(new_list[i].number_race);
}

for (int i = 0; i < 10 - 1; i++)
{
    for (int j = (10 - 1); j > i; j--) // для всех элементов после i-ого
    {
        if (arr[j - 1][1] > arr[j][1]) // если текущий элемент меньше
предыдущего
        {
            int temp[2];
            temp[0] = arr[j - 1][0]; // меняем их местами
            temp[1] = arr[j - 1][1]; // меняем их местами
            arr[j - 1][0] = arr[j][0];
            arr[j - 1][1] = arr[j][1];
            arr[j][0] = temp[0];
            arr[j][1] = temp[1];
        }
    }
}

for(int i=0; i<10; i++){
    cout << printMoreInformation(new_list[arr[i][0]]);
}

}

```