**Правительство Российской Федерации**

**Федеральное государственное автономное образовательное учреждение высшего профессионального образования   
"Национальный исследовательский университет   
"Высшая школа экономики"**

Московский институт электроники и математики Национального

исследовательского университета "Высшая школа экономики"

Департамент прикладной математики

**ОТЧЕТ**

**По лабораторной работе №8**

**По курсу «Алгоритмизация и программирование»**

|  |  |  |
| --- | --- | --- |
| ФИО студента | Номер группы | Дата |
| Колодин Матвей Алексеевич | БПМ213 | 01.02.2022 |
|  |
|  |
|  |

|  |
| --- |
|  |

**Москва – 2022 г.**

**ЗАДАНИЕ:**

**Сформировать массив записей имеющих заданную структуру (см. рис.) заданной**

**длины (не более 20). Вывести их на экран. Найти требуемую запись.**

**Общие требования:**

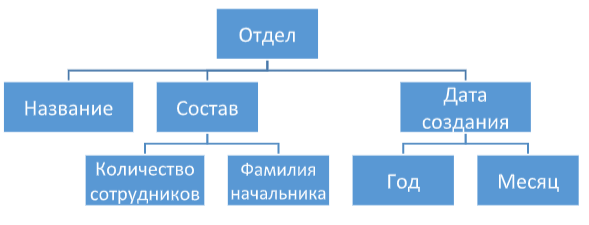
**1. Структуры определять в отдельных заголовочных файлах.**

**2. С помощью директив условной компиляции определить два способа ввода**

**исходного массива структур: пользователем с потока ввода и из заранее**

**заполненного массива.**

**(вариант №14)**





**РЕШЕНИЕ:**

#include <stdio.h>

#include <malloc.h>

#include <stdlib.h>

#define DEBUG 1

#define lenght 20

#include "otdel.c"

void input();

void output();

void findstruct();

int n;

struct otdel db[lenght];

int flg = 1;

int main()

{

system("chcp 1251");

system("cls");

input(); // ввод.

output(); // вывод.

if(flg != -1){

findstruct(); // отбор.

}

}

void input()

{

#if (DEBUG == 1)

printf("Введите количество отделов:\n");

scanf("%d", &n);

if(n <= 0){

printf("Результат проверки:\n");

printf("Отделов нет(");

flg = -1;

return 0;

}

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

// title:

printf("Введите название %d отдeла:\n", i+1);

scanf("\n"); // Нужно, для корректного считывания строк после таба.

gets(db[i].title.name\_dprt);

// composition:

printf("Введите состав %d отдела (количество работников и фамилию начальника):\n", i+1);

scanf("\n");

scanf("%d%s", &db[i].composition.nmb\_emp, &db[i].composition.sname\_chief);

//date:

printf("Введите дату создания %d отдела (год и месяц):\n", i+1);

scanf("%d", &db[i].date.year);

scanf("%d", &db[i].date.month);

}

#else

// отдел 1.

printf("Работа с заданным набором.\n");

strcpy(db[0].title.name\_dprt, "HSE");

db[0].composition.nmb\_emp = 1000;

strcpy(db[0].composition.sname\_chief, "Anisimov");

db[0].date.year = 1992;

db[0].date.month = 11;

// отдел 2

strcpy(db[1].title.name\_dprt, "MSU");

db[1].composition.nmb\_emp = 2000;

strcpy(db[1].composition.sname\_chief, "Sadovnichiy");

db[1].date.year = 1755;

db[1].date.month = 1;

n = 2;

#endif

}

void output()

{

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

printf("Отдeл номер, %d\nНазвание - %s \n", i+1, db[i].title.name\_dprt);

printf("(Состав %d отдела: %d работников, начальник - %s)\n", i+1, db[i].composition.nmb\_emp, db[i].composition.sname\_chief);

printf("Дата создания %d отдела - %d %d \n", i+1, db[i].date.year, db[i].date.month);

}

}

void findstruct()

{

// Вводим критерии проверки в заданном формате.

printf("Введите временный интервал, в котором будет осуществлен поиск отделов (гггг мм - гггг мм):\n");

int st\_m, st\_y, fn\_m, fn\_y;

scanf("%d %d - %d %d", &st\_y, &st\_m, &fn\_y, &fn\_m);

printf("Введите количество сотрудников ( минимальное количество - максимальное количество ):\n");

int cnt\_mn, cnt\_mx;

scanf("%d - %d", &cnt\_mn, &cnt\_mx);

// Начинаем обрабатывать массив структур)

printf("Результат проверки:\n");

int k = 0;

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

if(db[i].date.year >= st\_y && db[i].date.year <= fn\_y){

if(st\_y \* 12 + st\_m <= db[i].date.year \* 12 + db[i].date.month && fn\_y \* 12 + fn\_m >= db[i].date.year \* 12 + db[i].date.month){

if(db[i].composition.nmb\_emp >= cnt\_mn && db[i].composition.nmb\_emp <= cnt\_mx){

printf("Отдел %d подходит! Начальник - %s\n", i+1, db[i].composition.sname\_chief);

++k;

}

}

}

}

if(k == 0){

printf("Отделов нет(");

return 0;

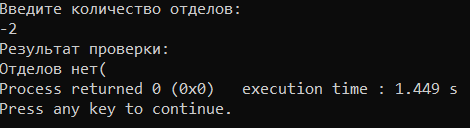
}

}

**ТЕСТЫ:**

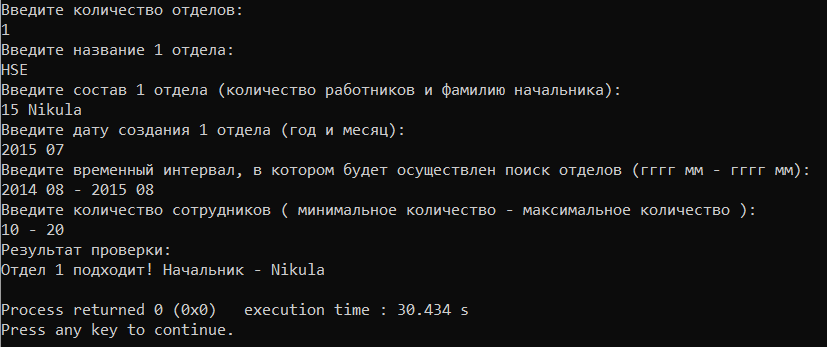
1. **Опять тесты идут в смешанном порядке – с ответом и без. Каждый подписан.**
2. **Будет один тест на заданный массив.**

**Тест № 1**



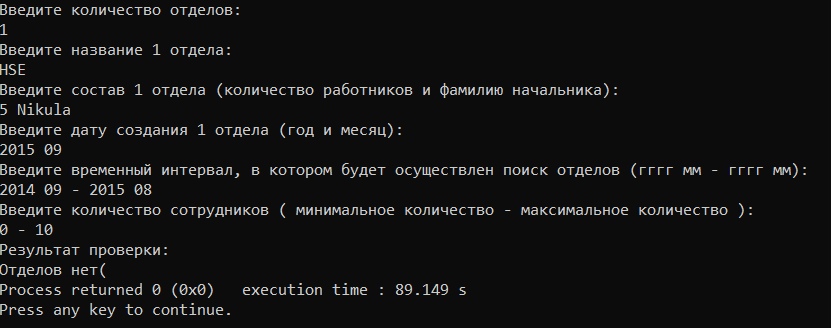
*(Введено некорректное число отделов.)*

**Тест № 2**



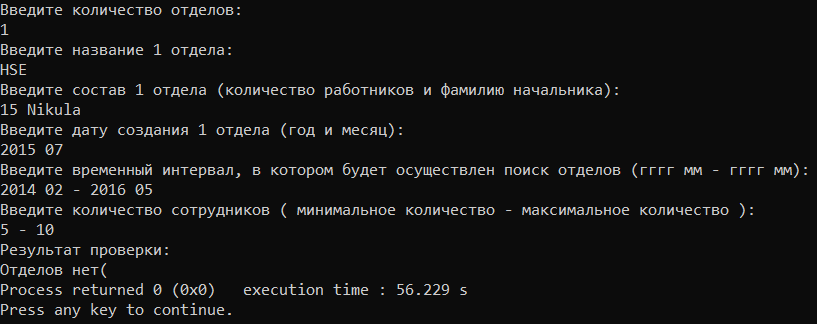
*(В тесте всего один отдел и он подходит по всем критериям.)*

**Тест № 3**



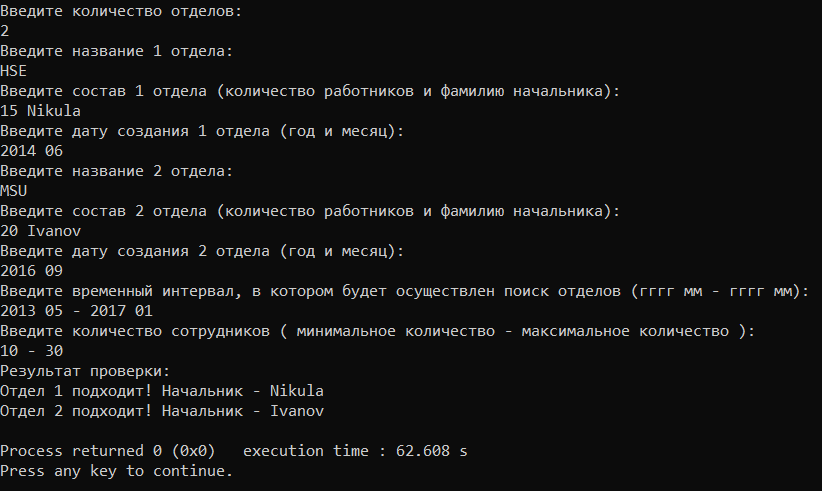
*(В тесте всего один отдел и он не подходит по дате.)*

**Тест № 4**



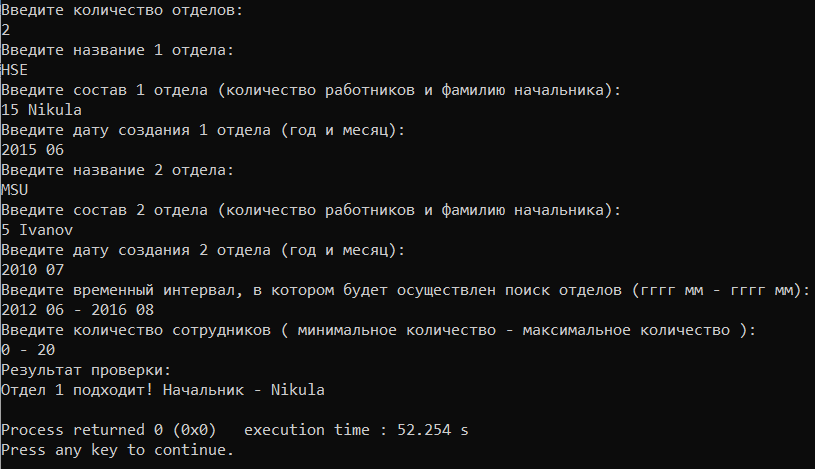
*(В тесте всего один отдел и он не подходит по числу сотрудников.)*

**Тест № 5**



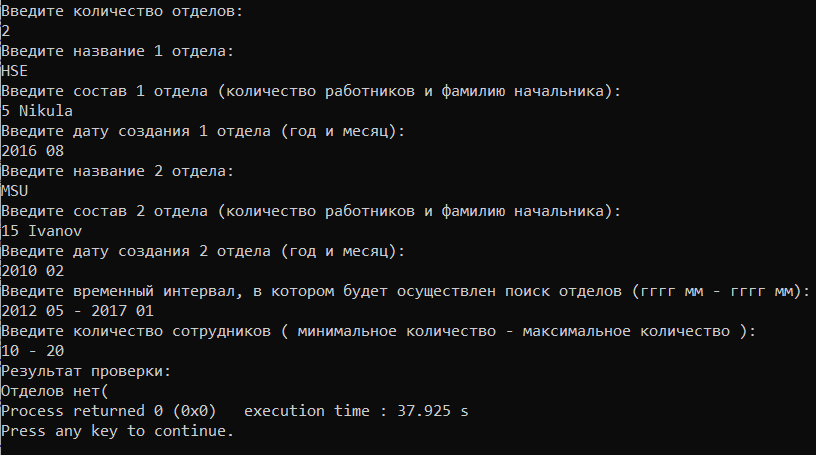
*(В тесте 2 отдела и оба подходят.)*

**(на следующей странице еще тесты)**

**Тест № 6** 

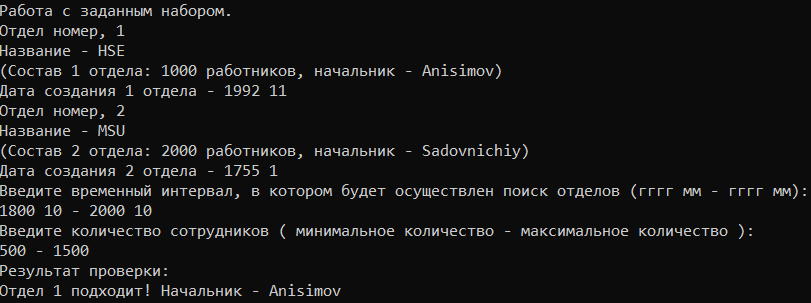
*(Первый отдел устраивает, а второй не подходит по временному интервалу.)*

**Тест № 7**



*(Первый отдел не подходит по количеству сотрудников, а второй по временному интервалу.)*

**Тест № 8**



*(Первый отдел подходит, а второй не подходит ни по одному из критериев. Мы работали с заданным набором данных.)*