Министерство науки и высшего образования Российской Федерации  
Пензенский государственный университет  
Кафедра вычислительная техника

**ОТЧËТ**  
по лабораторной работе №1  
по курсу «Логика и основы алгоритмизации в инженерных задачах»  
на тему «Простые структуры данных»

Выполнили студенты группы 22ВВВ2:  
Хоссейни Нежад С. А. С. М.

Приняли:  
Акифьев И. В.  
Митрохин М. А.

Пенза 2023

**Название**

Простые структуры данных

**Цель работы**

Изучение простых структур данных

**Лабораторное задание**

Задание 1: написать программу, вычисляющую разницу между максимальным и минимальным элементами массива.

Задание 2: написать программу, реализующую инициализацию массива случайными числами.

Задание 3: написать программу, реализующую создание массива произвольного размера, вводимого с клавиатуры.

Задание 4: написать программу, вычисляющую сумму значений в каждом столбце (или строке) двумерного массива.

Задание 5: написать программу, осуществляющую поиск среди структур student структуру с  заданными параметрами (фамилией, именем и т.д.).

**Метод решения задачи**

Заполнять массивы будем случайными числами, найдем разницу между максимальным и минимальным элементом массива с помощью дополнительных переменных, циклов и условий, сумму будем искать в строке двумерного массива, поиск студента будем осуществлять с помощью проверки в условии его имени и фамилии

**Листинг**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#include <string.h>

void excercise1();

void excercise2();

void excercise3();

void excercise4();

void excercise5();

int main()

{

srand(time(NULL));

excercise1();

excercise2();

excercise3();

excercise4();

excercise5();

return 0;

}

void excercise1()

{

int Arr[10] = {0};

int max = -1000;

int min = 1000;

printf("excercise 1:\n");

printf("Arr: ");

for (int i = 0; i < 10; i++)

{

Arr[i] = rand() % 100;

printf("%d ", Arr[i]);

}

for (int i = 0; i < 10; i++)

{

if (min > Arr[i])

{

min = Arr[i];

}

if (max < Arr[i])

{

max = Arr[i];

}

}

printf("\nmax = %d\nmin = %d", max, min);

printf("\ndifference: %d", max - min);

}

void excercise2()

{

int Arr[10] = { 0 };

for (int i = 0; i < 10; i++)

{

Arr[i] = rand() % 100;

}

printf("\n\n\nexcercise 2:\nArr: ");

for (int i = 0; i < 10; i++)

{

printf("%d ", Arr[i]);

}

}

void excercise3()

{

int\* Arr;

int n;

printf("\n\n\nexcercise 3:\nEnter the size of array: ");

scanf("%d", &n);

Arr = (int\*)malloc(n \* 4);

printf("Arr: ");

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

{

Arr[i] = rand() % 100;

printf("%d ", Arr[i]);

}

}

void excercise4()

{

int result = 0;

int Arr[3][3];

printf("\n\n\nexcercise 4:\n");

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 3; j++)

{

Arr[i][j] = rand() % 100;

result += Arr[i][j];

printf("%3d ", Arr[i][j]);

}

printf("result = %d", result);

result = 0;

printf("\n");

}

}

void excercise5()

{

typedef struct student

{

char name[20];

char secondName[20];

char facult[20];

char nomzach[20];

}Student;

char name[20];

char secondName[20];

Student student[3];

student[0] = { "Vanya", "Andreevich", "FVT", "555"};

student[1] = { "Petya", "Olegovich", "FITE", "333"};

student[2] = { "Andrey", "Raskolnikov", "FPTET"};

printf("\n\nexcercise 5:\n");

printf("\nEnter the student's first and last name: ");

scanf("%s %s", name, secondName);

for (int i = 0; i < 3; i++)

{

if (strcmp(student[i].name,name) == 0 and strcmp(student[i].secondName,secondName) == 0)

{

printf("\nStudent %s %s number %d, studies at the faculty %s with record book number %s\n", student[i].name, student[i].secondName, i, student[i].facult, student[i].nomzach);

}

}

}

**Пояснительный текст к программе**

В 1 задании мы сначала заполняем массив, а после находим максимальный и минимальный элемент и их разницу.

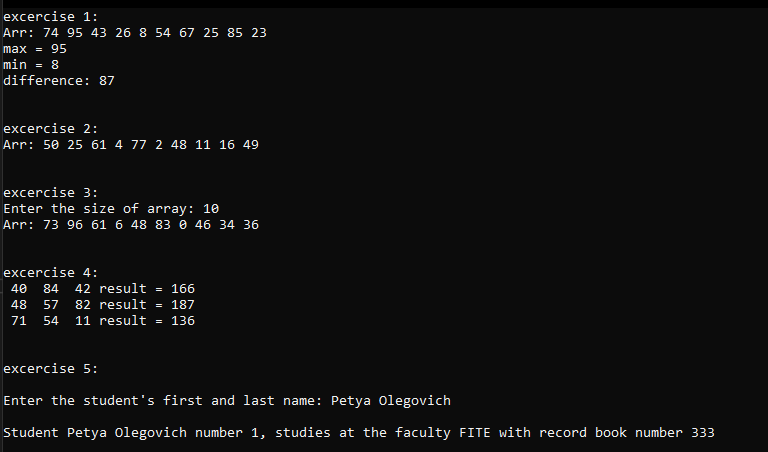
Во 2 задании используем rand() % 100, т.е берем остаток от деления на 100, чтобы получать числа меньше 100.

В 3 задании используем функцию для выделения памяти malloc().

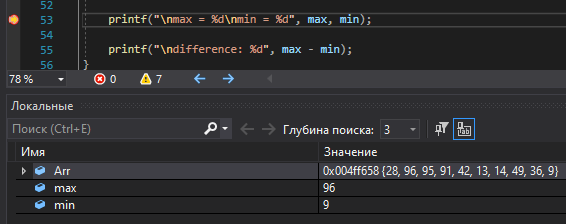
В 4 задании i – строки, поэтому в цикле j мы суммируем строку и после обнуляем сумму.

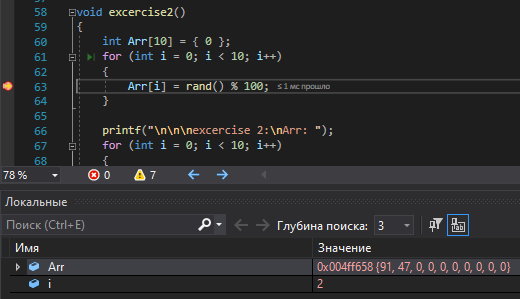
В 5 задании мы осуществляем поиск по имени и фамилии.

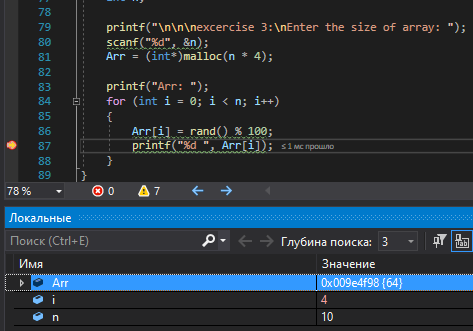
**Результат работы программы**

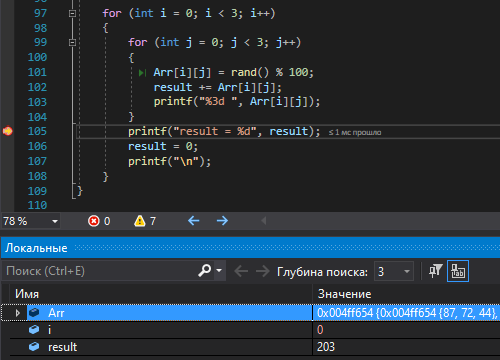
****

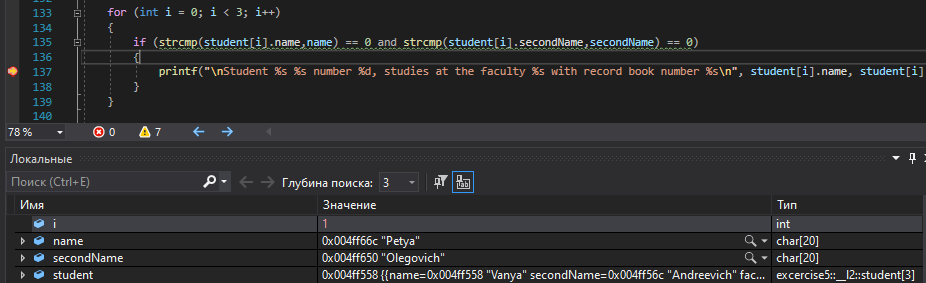
**Протокол трассировки программы**

****

****

****

****

****

**Вывод**

В результате выполнения лабораторной работы я изучил простые структуры данных.