Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе № 1

по курсу «Логика и основы алгоритмизации в инженерных задачах»

на тему «Простые структуры данных»

Выполнил

студент группы 22ВВВ2:

Зубриянова А.А.

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.).

**Описание метода решения задачи**

a) По Заданию 2 создаем массив а[5][5] и инициализируем его случайными числам, использую функции rand и srand.

По Заданию 1 находим разницу между максимальным и минимальным элементами массива: для этого присваиваем переменным, которые будут хранить максимальное и минимальное значения (max и min), значение первого элемента массива. Затем в цикле перебираем все элементы массива, сравнивая их с max и min и в случае нахождения большего (меньшего) числа, чем max (min), присваиваем его значение max (min). Находим разницу max и min.

По Заданию 4 находим сумму значений каждой строки двумерного массива а[5][5]. Храним сумму в переменной sum, обнуляя ее каждый раз перед вложенным циклом for для накопления суммы каждой отдельной строки.

б) По Заданию 3 с клавиатуры вводится массив структур a произвольного размера с использованием выделения памяти через функцию malloc. Задание 5. Структура Student содержит поля num (номер зачетки), firstname, name, faculty. Поля заполняются в цикле и затем выводятся. После пользователь вводит номер зачетки (переменная m), чтобы найти нужного студента (определенную структуру). Если m совпадает с num, выводится вся информация для этой зачетки.

**Листинг**

а) **lab1. cpp**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <locale.h>

#include <stdlib.h>

#include <time.h>

int main()

{

setlocale(LC\_ALL, "Russian");

//srand(time(NULL));

int a[5][5];

//Задание 2. Инициализация массива случайными числами

printf("Задание 2. Инициализация массива случайными числами\n");

for (int i = 0; i < 5; i++)

{

for (int j = 0; j < 5; j++)

{

a[i][j] = rand() - rand();

printf("%8d ", a[i][j]);

}

printf("\n");

}

printf("\n");

//Задание 1. Разница между макс и мин элементами массива

printf("Задание 1. Разница между макс и мин элементами массива\n");

int max = a[0][0];

int min = a[0][0];

for (int i = 0; i < 5; i++)

{

for (int j = 0; j < 5; j++)

{

if (max < a[i][j])

{

max = a[i][j];

}

if (min > a[i][j])

{

min = a[i][j];

}

}

}

int d = max - min;

printf("max = %d, min = %d, разность = %d\n", max, min, d);

printf("\n");

//Задание 4. Сумма значений каждой строки двумерного массива

printf("Задание 4. Сумма значений каждой строки двумерного массива\n");

int sum;

for (int i = 0; i < 5; i++)

{

sum = 0;

for (int j = 0; j < 5; j++)

{

sum += a[i][j];

}

printf("строка %d = %d \n", i, sum);

}

return 0;

}

б) **lab1\_2. cpp**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <locale.h>

#include <stdlib.h>

#include <time.h>

#include <windows.h>

struct Student

{

char num[5];

char firstname[20];

char name[20];

char faculty[20];

};

int main()

{

setlocale(LC\_ALL, "Russian");

srand(time(NULL));

//Задание 3 + Задание 5. Ввод массива произвольного размера и поиск структур

printf("Задание 3 + Задание 5. Ввод массива произвольного размера и поиск структур\n");

int n;

printf("Введите количество студентов:");

scanf("%d", &n);

Student\* a;

a = (Student\*)malloc(n \* sizeof(Student));

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

{

//a[i].num = 100 + i;

snprintf(a[i].num, 5, "%d", i);

snprintf(a[i].firstname, 20, "Фамилия%d", i);

snprintf(a[i].name, 20, "Имя%d", i);

snprintf(a[i].faculty, 20, "Факультет%d", i);

}

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

{

printf("%5s", a[i].num);

printf("%15s", a[i].firstname);

printf("%15s", a[i].name);

printf("%20s", a[i].faculty);

printf("\n");

}

char m[5];

printf("\n");

printf("Введите номер зачетки: ");

scanf("%s", m);

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

{

if (strcmp(a[i].num, m) == 0)

{

printf("%5s", a[i].num);

printf("%15s", a[i].firstname);

printf("%15s", a[i].name);

printf("%20s", a[i].faculty);

printf("\n");

}

}

int l = 0;

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

{

if (strcmp(a[i].num, m) != 0)

{

l++;

}

}

if (l == n)

{

printf("Не существует такой зачетки ");

}

free(a);

return 0;

}

**Результаты работы программы**

Результаты работы программы представлены на рисунках:

а)

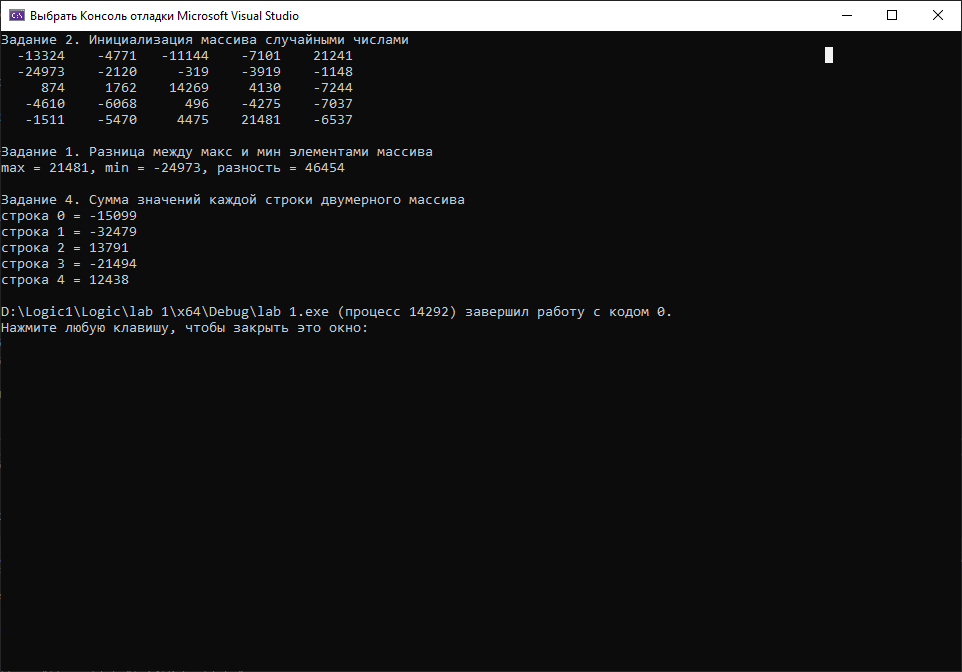


Рисунок 1 – Задания 1, 2, 4

б)

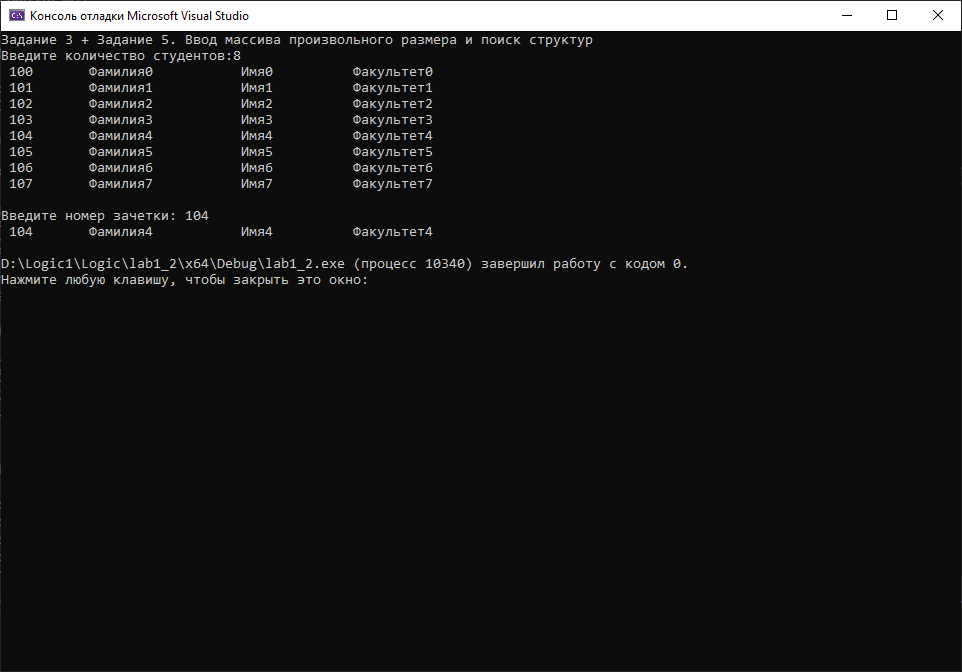


Рисунок 2 – Задания 3, 5

**Вывод**

В ходе выполнения лабораторной работы была разработана программа с использованием простых структур данных. Результаты работы программы совпали с ожидаемыми результатами, следовательно, программа работает без ошибок.

Получили опыт в создании проектов в среде Microsoft Visual Studio, приобрели навыки программирования алгоритмов.