Пензенский государственный университет

Кафедра "Вычислительная техника"

**ОТЧЕТ**

по дисциплине "Логика и основы алгоритмизации в инженерных задачах"

по лабораторной работе №1

на тему "Простые структуры данных"

Выполнили студенты гр. 21ВВ4:

Савкин В.В.

Савостин Е.Е.

Проверили:

Юрова О.В.

Акифьев И.В.

Пенза, 2022

**Цель работы**: Повторить способы создания и заполнения массивов и структур.

**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.).

**Ход работы:**

**Библиотеки:**

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#include <malloc.h>

#include <string.h>

#include <windows.h>

**Задание 1,2,3:**

setlocale(LC\_ALL, "RUS");

srand(time(NULL));

int k;

printf("Введите размер массива");

scanf\_s("%d", &k);

int\* arr = (int\*)malloc(k \* sizeof(int));// размер первого массива

int min, max, razn;

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

{

arr[i] = rand() % 100;

}

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

{

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

}

min = arr[0], max = arr[0];

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

{

if (min > arr[i])

{

min = arr[i];

}

if (max < arr[i])

{

max = arr[i];

}

}

razn = max - min;

printf("\nРазница между максимальным и минимальным элементом массива = %d\n", razn);

**Задание 4:**

#define N 4

#define M 5

Int main()

{

int sum\_r[N] = { 0, 0, 0, 0 };

int sum\_c[M] = { 0, 0, 0, 0, 0 };

int\*\* matr;

int g, h;

matr = (int\*\*)malloc(N \* M \* sizeof(int\*));

for (g = 0; g < N; ++g)

matr[g] = (int\*)malloc(M \* sizeof(int));

for (g = 0; g < N; ++g)

for (h = 0; h < M; ++h)

matr[g][h] = rand() % 50;

printf("2D Массив:\n");

for (g = 0; g < N; ++g)

{

for (h = 0; h < M; ++h)

{

printf("%d ", matr[g][h]);

}

printf("\n");

}

for (g = 0; g < N; ++g)

for (h = 0; h < M; ++h)

sum\_r[g] += +matr[g][h];

for (g = 0; g < M; ++g)

for (h = 0; h < N; ++h)

sum\_c[g] = sum\_c[g] + matr[h][g];

for (g = 0; g < N; ++g)

printf("Сумма %d строки: %d\n", g + 1, sum\_r[g]);

printf("\n");

for (g = 0; g < M; ++g)

printf("Сумма %d столбца: %d\n", g + 1, sum\_c[g]);

for (g = 0; g < N; ++g)

free(matr[g]);

free(matr); // освобождение памяти

return 0;

}

**Задание 5:**

#include <stdio.h>

#include <string.h>

#include <malloc.h>

#include <locale.h>

//Структура, описывающая студента

struct Student

{

char lastname[30];

char group[10];

};

//Функция формирует строковый массив фамилий студентов, содержащихся в структуре "students", которsе учатся в группе group и их фамилии начинаются на букву "С"

void surname(Student students[], int N, char group[10], char c)

{

//Посчитаем кол-во студентов, которое нужно будет вывести. Это будет число k

int k = 0;

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

{

//Если группа i-го студента равна group и первая буква его фамилии равна "C", то k++

if ((strcmp(students[i].group, group) == 0) && (students[i].lastname[0] == c))

{

k++;

}

}

//Создадим массив под k фамилий

char\*\* lastNames = (char\*\*)malloc(k \* sizeof(char\*));

k = 0;

//Занесём в массив "lastNames" все подходящие фамилии

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

{

if ((strcmp(students[i].group, group) == 0) && (students[i].lastname[0] == c))

{

lastNames[k++] = students[i].lastname;

}

}

//Выведем их в консоль

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

{

printf("%s\n", lastNames[i]);

}

//Освободим память из под массива lastNames

free(lastNames);

}

int main()

{

setlocale(LC\_ALL, "RUS");

//Объявим массив структур и инициализируем его данными

Student students[6] = {

"Ivanov","21vv-4",

"Savkin","21vv-4",

"Savostin","21vv-4",

"Snegirev","21vv-4",

"Petrov","21vv-4",

"Dugerenko","21vv-4",

};

//Сюда сохраним название группы из консоли

char group[10];

//Читаем название группы из консоли

printf("Введите название группы \n");

gets\_s(group);

printf("Введите первую букву фамилии студента \n");

//Читаем из консоли первую букву фамилии

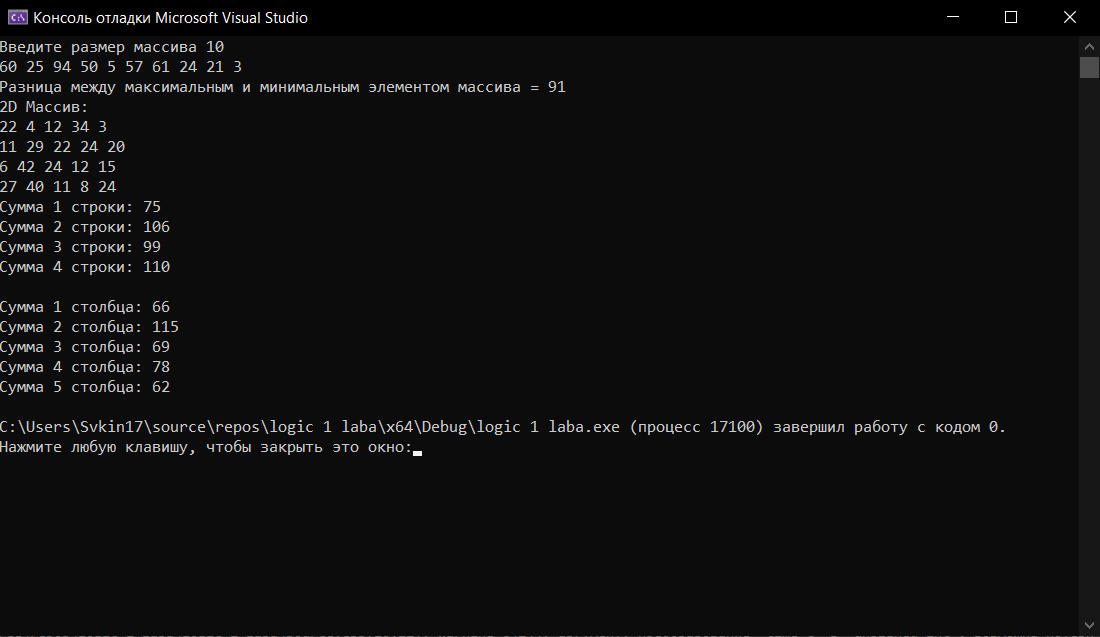
char c = getchar();

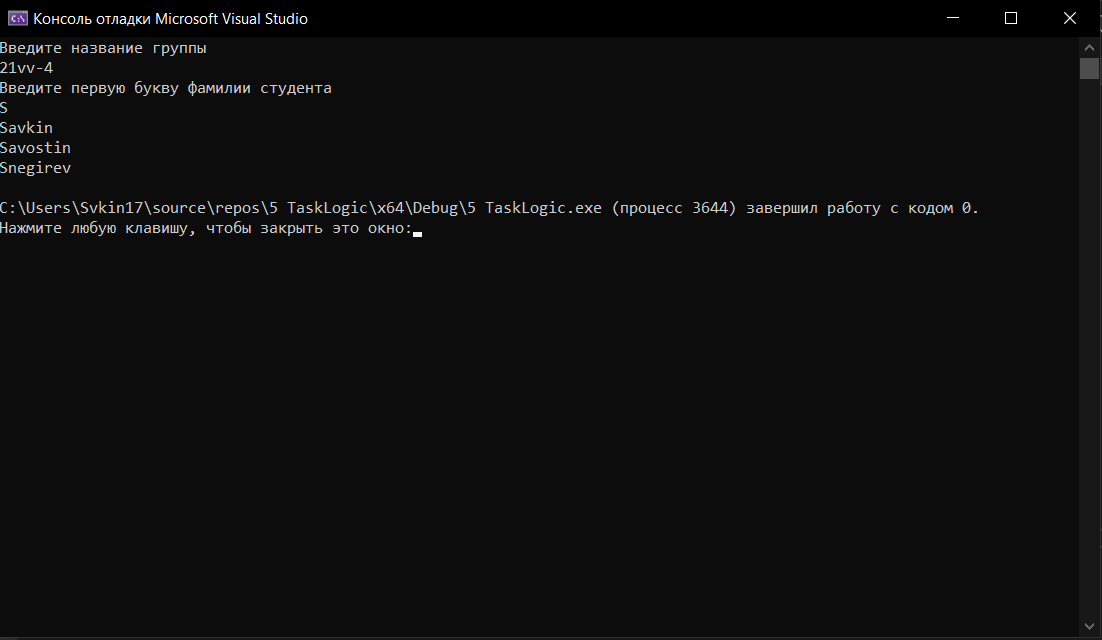
//Вызываем саму функцию

surname(students, 6, group, c);

}

**Результат работы программ:**

****

****

**Вывод:** Мы повторили способы задания массивов и структур.