**Министерство науки и высшего образования Российской Федерации**

**Пензенский государственный университет**

**Кафедра “Вычислительная техника”**

ОТЧЁТ

По лабораторной работе №1

По курсу «Логика и основы алгоритмизации в инженерных задачах»

На тему «Простые структуры данных»

Вариант №1

Выполнил студенты группы 23ВВВ4:

Зарубин Я.Д.\_\_\_\_

Герасимов К.Б.\_\_\_\_

Приняли:

к.т.н. доцент Деев М.В. \_\_\_\_

к.т.н. доцент Юрова О.В. \_\_\_\_

Пенза 2024

**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.).

### Листинг

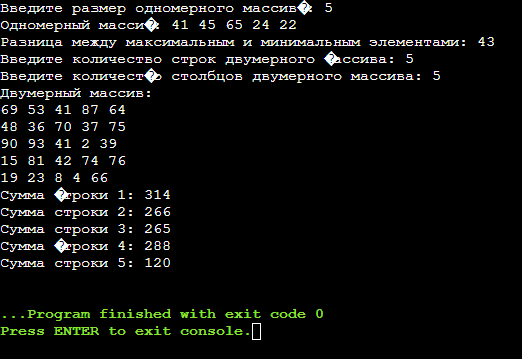
**Задания 1-4)**

[#include](https://vk.com/im?sel=359740135&st=%23include) <stdio.h>  
[#include](https://vk.com/im?sel=359740135&st=%23include) <stdlib.h>  
[#include](https://vk.com/im?sel=359740135&st=%23include) <time.h>  
  
// Задание 1: Вычисление разницы между максимальным и минимальным элементами массива  
int findDifference(int [\*array](https://vk.com/id20026727), int size) {  
int max = array[0];  
int min = array[0];  
  
for(int i = 1; i < size; i++) {  
if(array[i] > max)  
max = array[i];  
if(array[i] < min)  
min = array[i];  
}  
  
return max - min;  
}  
  
// Задание 2: Инициализация массива случайными числами  
void initializeArray(int [\*array](https://vk.com/id20026727), int size) {  
for(int i = 0; i < size; i++) {  
array[i] = rand() % 101;  
}  
}  
  
// Задание 4: Вычисление суммы значений в каждой строке двумерного массива  
void sumRows(int \*\*array, int rows, int cols) {  
for(int i = 0; i < rows; i++) {  
int sum = 0;  
for(int j = 0; j < cols; j++) {  
sum += array[i][j];  
}  
printf("Сумма строки %d: %d\n", i + 1, sum);  
}  
}  
  
int main() {  
srand(time(NULL));  
  
// Задание 3: Создание массива произвольного размера  
int size;  
printf("Введите размер одномерного массива: ");  
scanf("%d", &size);  
  
int [\*array](https://vk.com/id20026727) = (int\*)malloc(size \* sizeof(int));  
int lower\_bound, upper\_bound;  
  
// Инициализация и вывод одномерного массива  
initializeArray(array, size);  
printf("Одномерный массив: ");  
for(int i = 0; i < size; i++) {  
printf("%d ", array[i]);  
}  
printf("\n");  
  
// Вычисление разницы между максимальным и минимальным элементами  
int difference = findDifference(array, size);  
printf("Разница между максимальным и минимальным элементами: %d\n", difference);  
  
// Работа с двумерным массивом  
int rows, cols;  
printf("Введите количество строк двумерного массива: ");  
scanf("%d", &rows);  
printf("Введите количество столбцов двумерного массива: ");  
scanf("%d", &cols);  
  
int \*\*array2D = (int\*\*)malloc(rows \* sizeof(int\*));  
for(int i = 0; i < rows; i++) {  
array2D[i] = (int\*)malloc(cols \* sizeof(int));  
initializeArray(array2D[i], cols);  
}  
  
// Вывод двумерного массива  
printf("Двумерный массив:\n");  
for(int i = 0; i < rows; i++) {  
for(int j = 0; j < cols; j++) {  
printf("%d ", array2D[i][j]);  
}  
printf("\n");  
}  
  
// Вычисление суммы значений в каждой строке двумерного массива  
sumRows(array2D, rows, cols);  
  
// Освобождение выделенной памяти  
free(array);  
for(int i = 0; i < rows; i++) {  
free(array2D[i]);  
}  
free(array2D);  
  
return 0;  
}

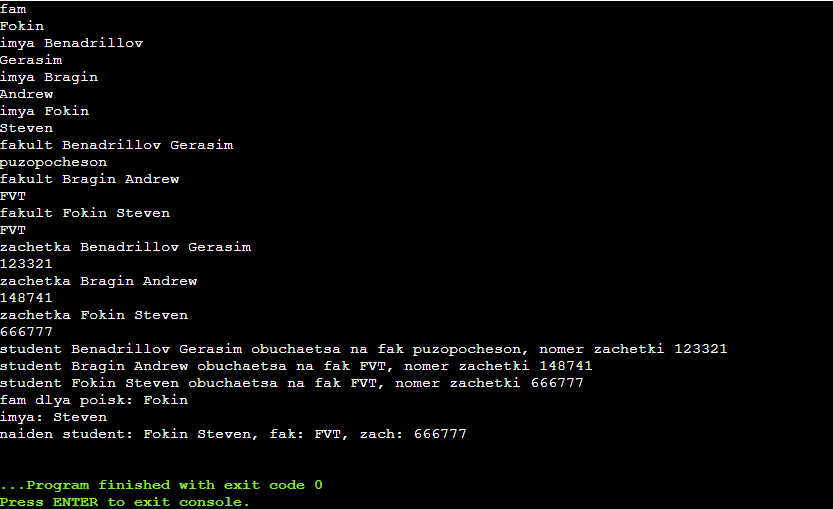
**Задание 5)**

[#define](https://vk.com/im?sel=359740135&st=%23define) \_CRT\_SECURE\_NO\_WARNINGS  
[#include](https://vk.com/im?sel=359740135&st=%23include) <stdio.h>  
[#include](https://vk.com/im?sel=359740135&st=%23include) <string.h>  
  
int main(void) {  
setvbuf(stdin, NULL, \_IONBF, 0);  
setvbuf(stdout, NULL, \_IONBF, 0);  
  
int i;  
struct student {  
char famil[20];  
char name[20];  
char facult[20];  
int Nomzach;  
} stud[3];  
  
// Ввод данных о студентах  
for (i = 0; i < 3; i++) {  
printf("fam\n");  
scanf("%19s", stud[i].famil);  
}  
for (i = 0; i < 3; i++) {  
printf("imya %s\n", stud[i].famil);  
scanf("%19s", stud[i].name);  
}  
for (i = 0; i < 3; i++) {  
printf("fakult %s %s\n", stud[i].famil, stud[i].name);  
scanf("%19s", stud[i].facult);  
}  
for (i = 0; i < 3; i++) {  
printf("zachetka %s %s\n", stud[i].famil, stud[i].name);  
scanf("%d", &stud[i].Nomzach);  
}  
  
// Вывод данных о студентах  
for (i = 0; i < 3; i++) {  
printf("student %s %s obuchaetsa na fak %s, nomer zachetki %d\n",  
stud[i].famil, stud[i].name, stud[i].facult, stud[i].Nomzach);  
}  
  
// Задание 5: поиск студента по заданным параметрам  
char search\_famil[20], search\_name[20];  
printf("fam dlya poisk: ");  
scanf("%19s", search\_famil);  
printf("imya: ");  
scanf("%19s", search\_name);  
  
int found = 0; // Флаг для проверки, найден ли студент  
  
for (i = 0; i < 3; i++) {  
if (strcmp(stud[i].famil, search\_famil) == 0 && strcmp(stud[i].name, search\_name) == 0) {  
printf("naiden student: %s %s, fak: %s, zach: %d\n",  
stud[i].famil, stud[i].name, stud[i].facult, stud[i].Nomzach);  
found = 1;  
break;  
}  
}  
  
if (!found) {  
printf("stud s fam %s i name %s not naiden\n", search\_famil, search\_name);  
}  
  
return 0;  
}

**Результат работы программы**



**Рисунок 1 — Результаты работы программы с заданиями 1-4.**



**Рисунок 2 — Результаты работы программы с заданиями 5.**