Пензенский государственный университет  
Кафедра «Вычислительной техники»

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

**Выполнили студенты группы 21ВВП2:**

Миликов А.Р.

Большаков Е.К.

**Приняли:**

д.т.н. профессор Митрохин М. А.

Юрова О.В.

Пенза 2023

**Цель работы:** вспомнить работу с массивами и структурами.

**Теория**

**Массив** – это конечная совокупность данных одного типа.  
Массивы могут состоять из целых чисел, чисел с плавающей запятой, символов и других типов данных. Существуют даже массивы массивов, – многомерные массивы. Любой массив в программе должен быть описан: вначале указывается тип массива, затем его имя, после имени массива добавляют квадратные скобки ([ ]) внутри которых обычно стоит число, показывающее количество элементов массива.

**Динамический массив**

Для использования функций динамического распределения памяти необходимо подключение библиотеки <malloc.h> или <stdlib.h> (в зависимости от компилятора и используемого стандарта языка).

Функции динамического выделения памяти

void\* malloc(*РазмерМассиваВБайтах*);  
void\* calloc(*ЧислоЭлементов*, *РазмерЭлементаВБайтах*);

Элементами массива могут быть как данные простых типов, так и составных типов (например, структуры), поэтому для точного определения размера элемента массива в общем случае используется функция

int sizeof(*ТипЭлемента*);

Память, динамически выделенная с использованием функций calloc(), malloc(), должна быть освобождена после окончания использования функцией

free(*указатель*);

**Структура** - это совокупность переменных одного или нескольких типов, сгруппированных в один элемент.

Объявление структуры осуществляется с помощью ключевого слова struct, за которым идет ее тип и далее список элементов, заключенных в фигурные скобки:

struct тип {   
тип элемента\_1 имя элемента\_1;

                    .........

тип элемента\_n имя элемента\_n;

 };

**Листинг**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <conio.h>

#include <stdlib.h>

#include <stdio.h>

#include <time.h>

#include <iostream>

int main(void)

{

int mas1[5] = {10, 12, 2, -3, 5};

int max, min;

min = mas1[0] ;

max = mas1[0] ;

printf("ZADANIE 1\n");

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

printf("%3d| ", mas1[i]);

}

for (int i = 1; i < 5; i++) {

if (mas1[i] > max)

max = mas1[i];

if (mas1[i] < min)

min = mas1[i];

}

printf("\nmax:%2d\nmin:%2d\nRaznica: %2d", max, min, max-min);

printf("\n\nZADANIE 2\n");

int mas2[10];

srand(time(0));

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

mas2[i] = rand() % 100;

}

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

printf("%3d|", mas2[i]);

}

printf("\n\nZADANIE 3\n");

int \*mas3, k;

printf("Vvedite razmer massiva: ");

scanf("%d", &k);

mas3 = (int\*)malloc(k \* sizeof(int));

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

mas3[i] = rand() % 100 - 10;

}

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

printf("%3d|", mas3[i]);

}

free(mas3);

printf("\n\nZADANIE 4\n");

int mas4[3][3], ss, st;

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

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

mas4[i][j] = rand() % 10;

}

}

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

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

printf("%d ", mas4[i][j]);

}

printf("\n");

}

printf("Summa strok: ");

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

ss = 0;

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

ss += mas4[i][j];

}

printf("%d ", ss);

}

printf("\n");

printf("Summa stolbcov: ");

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

st = 0;

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

st += mas4[i][j];

}

printf("%d ", st);

}

printf("\n\nZADANIE 5\n");

struct student {

int numzach;

char familia[20];

char name[20];

char facult[20];

} stud[3], search;

int i, flag = 0;

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

printf("Vvedite familiu studenta: ");

scanf("%20s", &stud[i].familia);

}

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

printf("Vvedite imya studenta: ");

scanf("%20s", &stud[i].name);

}

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

printf("Vvedite nazvanie faculteta studenta %s: ", stud[i].familia);

scanf("%20s", &stud[i].facult);

}

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

printf("Vvedite nomer zachetnoy knijki studenta %s: ", stud[i].familia);

scanf("%d", &stud[i].numzach);

}

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

printf("Student %s %s obychaetsya na fakultete %s, nomer zachetnoy knijki %d\n", stud[i].familia, stud[i].name, stud[i].facult, stud[i].numzach);

}

printf("\nPOISK\n");

printf("Vvedite familiu nujnogo studenta: ");

scanf("%s", &search.familia);

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

if (strcmp(stud[i].familia, search.familia) == 0) {

printf("Student %s %s obychaetsya na fakultete %s, nomer zachetnoy knijki %d\n", stud[i].familia, stud[i].name, stud[i].facult, stud[i].numzach);

flag++;

}

}

getchar();

}

**Результат работы программы**

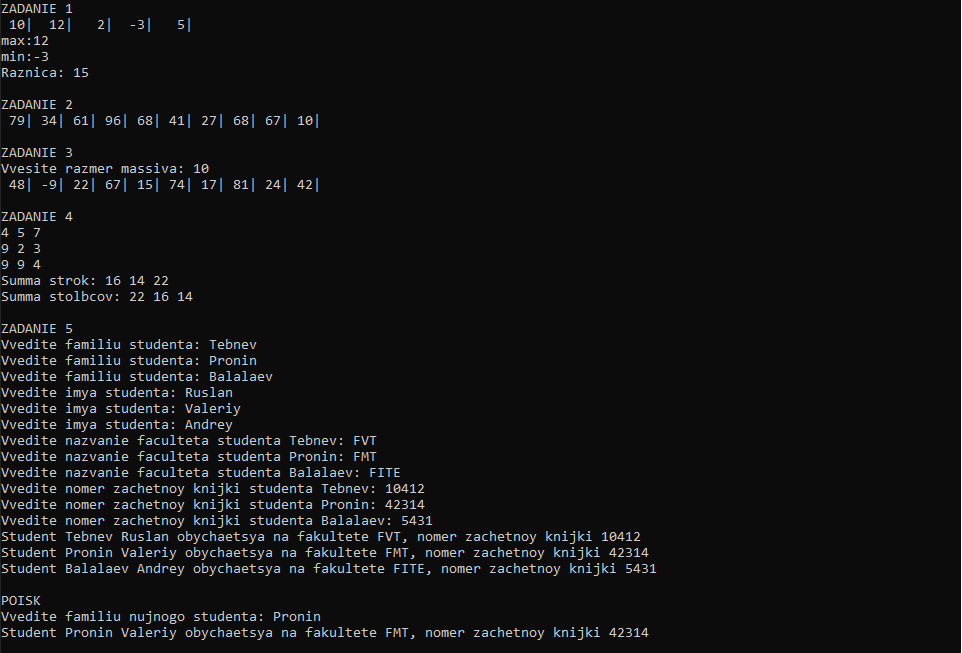


Рисунок 1 – Результат работы программы