1Министерство 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»

Выполнили:

студент группы 20ВВ2

Александров В. С.

Принял:

д.т.н., Митрохин М. А.

Пенза 2020

**Цель работы:**  Вспомнить простые структуры 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.).

**Листинг:**

#define \_CRT\_SECURE\_NO\_WARNINGS

#pragma warning(disable:4996)

#include <iostream>

#include <stdio.h>

#include <string.h>

#include <cstdlib>

#include <windows.h>

#include <conio.h>

#include <ctime>

#include <cmath>

int zadanie1() {

printf("Задание 1\n");

int b = 0;

int a[10] = { 1,2,3,4,5,6,7,8,9,10 };

int c = a[9];

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

{

if (a[i] > b)

{

b = a[i];

}

if (a[i] < c)

{

c = a[i];

}

};

printf("%d\n", (b - c));

return(0);

}

int zadanie2() {

printf("\nЗадание 2\n");

int a[10];

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

{

a[i] = rand() % 100;

printf("%d\n", a[i]);

}

return(0);

}

int zadanie3() {

int\*\* a;

int ii, znach, j;

printf("\nЗадание 3\nВведите размер массива: ");

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

a = (int\*\*)malloc(znach \* sizeof(int));

for (ii = 0; ii < znach; ii++)

{

a[ii] = (int\*)malloc(znach \* sizeof(int));

}

for (ii = 0; ii < znach; ii++)

{

for (j = 0; j < znach; j++)

{

// rand()%(M2-M1+1)+M1

a[ii][j] = rand() % (0 + 100 + 1) - 100;

printf("%3d", a[ii][j]);

}

printf("\n");

}

return(0);

}

int zadanie4() {

printf("\nЗадание 4\n");

int a[5][5], i = 0, j = 0, sum = 0;

for (i = 0; i < 4; i++) {

for (j = 0; j < 4; j++) {

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

}

}

for (i = 0; i < 4; i++) {

for (j = 0; j < 4; j++) {

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

}

printf("\n");

}

for (j = 0; j < 4; j++) {

for (i = 0; i < 4; i++) {

sum += a[i][j];

}

printf("сумма по столбцу %d:", j + 1);

printf("%3d ", sum);

printf("\n");

sum = 0;

}

printf("\n");

for (i = 0; i < 4; i++) {

for (j = 0; j < 4; j++) {

sum += a[i][j];

}

printf("сумма по строке %d:", i + 1);

printf("%3d ", sum);

printf("\n");

sum = 0;

}

return(0);

}

struct student {

char surname[100];

char name[100];

char faculty[10];

int kurs;

};

int zadanie5() {

printf("\nЗадание 5\n\n");

struct student b[10];

int i = 0, n = 3, search\_num;

char search[100], \* str;

int s[1];

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

printf("введите Фамилию: ");

scanf("%s", &b[i].surname);

//gets\_s(b[i].surname);

if (b[i].surname[0] == '\*') {

n = i;

break;

}

printf("введите имя: ");

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

printf("введите факульет: ");

scanf("%s", &b[i].faculty);

printf("введите курс: ");

scanf("%u", &b[i].kurs);

printf("\n");

gets\_s(b[9].faculty);

}

printf("%5.5s %15.15s %15.15s %25.25s %15.15s \n", "#", "фамилия", "имя", "факультет", "курс");

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

printf(" %5u %15.15s %15.15s %15.15s %25.u \n", i + 1, b[i].surname, b[i].name, b[i].faculty, b[i].kurs);

}

printf("\nНайти:\n ");

scanf("%s", &search);

printf("%5.5s %15.15s %15.15s %25.25s %15.15s \n", "#", "фамилия", "имя", "факультет", "курс");

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

str = strstr(b[i].surname, search);

if (str == NULL) {

}

else {

printf(" %5u %15.15s %15.15s %15.15s %25.u \n", i + 1, b[i].surname, b[i].name, b[i].faculty, b[i].kurs);

}

str = strstr(b[i].name, search);

if (str == NULL) {

}

else {

printf(" %5u %15.15s %15.15s %15.15s %25.u \n", i + 1, b[i].surname, b[i].name, b[i].faculty, b[i].kurs);

}

str = strstr(b[i].faculty, search);

if (str == NULL) {

}

else {

printf(" %5u %15.15s %15.15s %15.15s %25.u \n", i + 1, b[i].surname, b[i].name, b[i].faculty, b[i].kurs);

}

}

return(0);

}

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

zadanie1();

zadanie2();

zadanie3();

zadanie4();

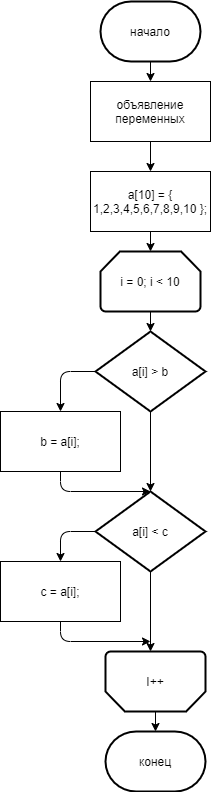
zadanie5();

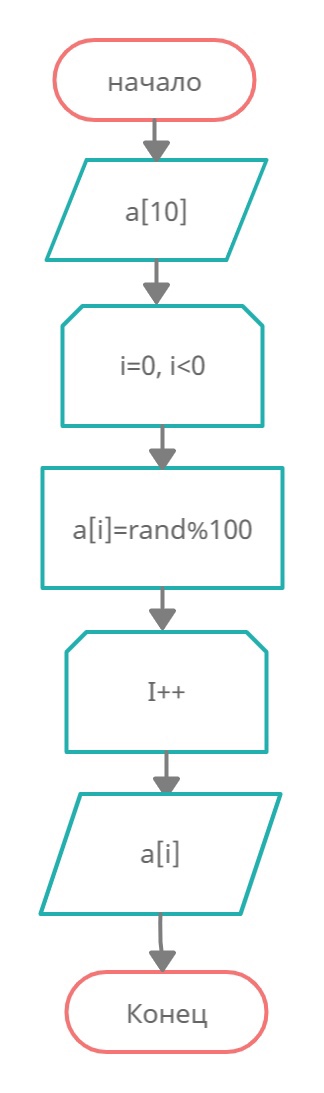
}

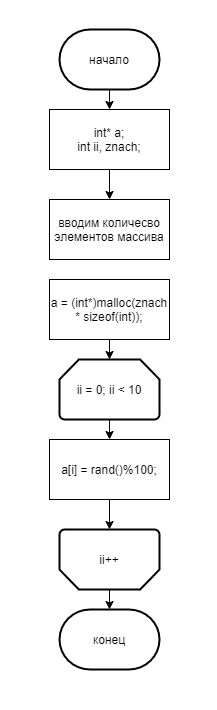
**Пояснительный текст к программе:**

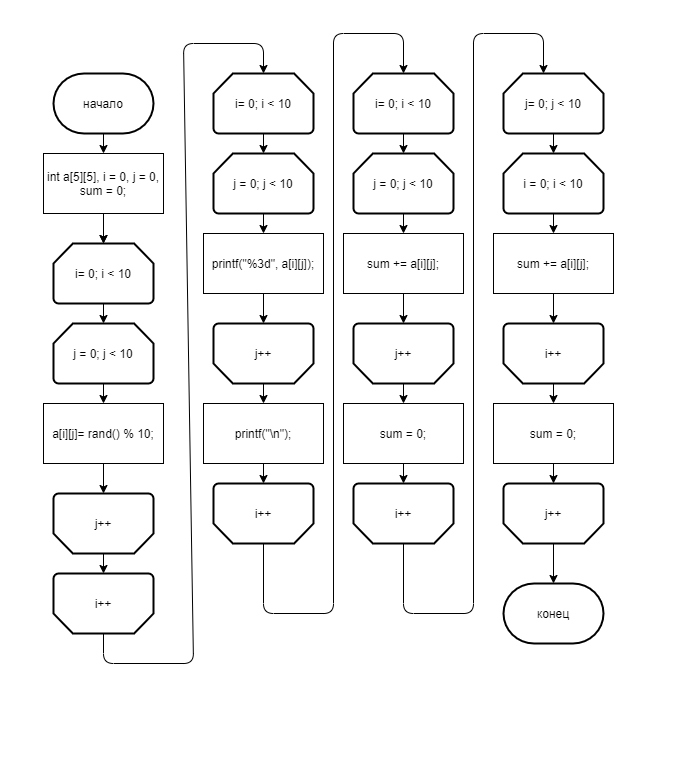
1. В переменную surname[100];записывается фамилия
2. В переменную name[100]записывается имя
3. В переменную faculty[10];записывается факультет
4. В переменную kurs записывается нормер курса
5. Переменная **i**  это сётчик цикла
6. **n** это количество повторений цикла
7. **b[10]** это ссылка на структуру student

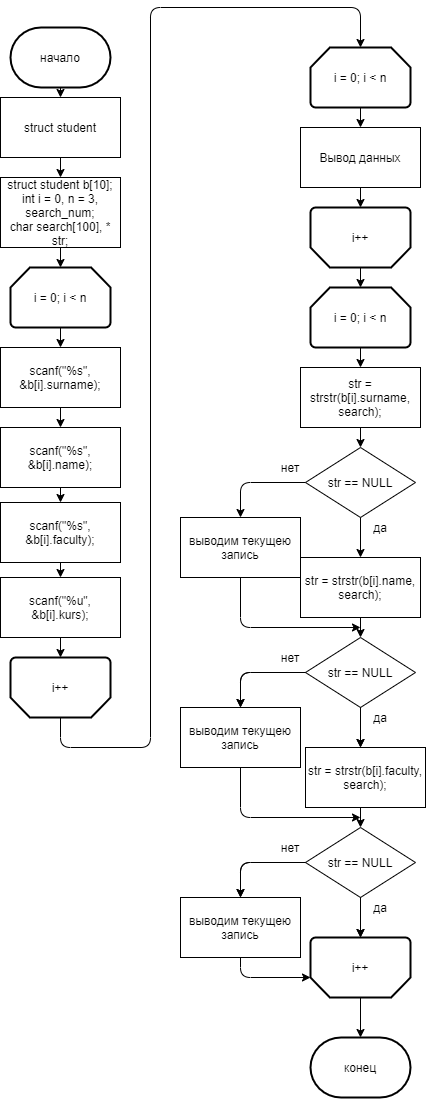
**Схема программы:**

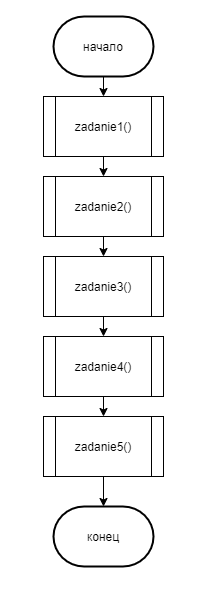
****



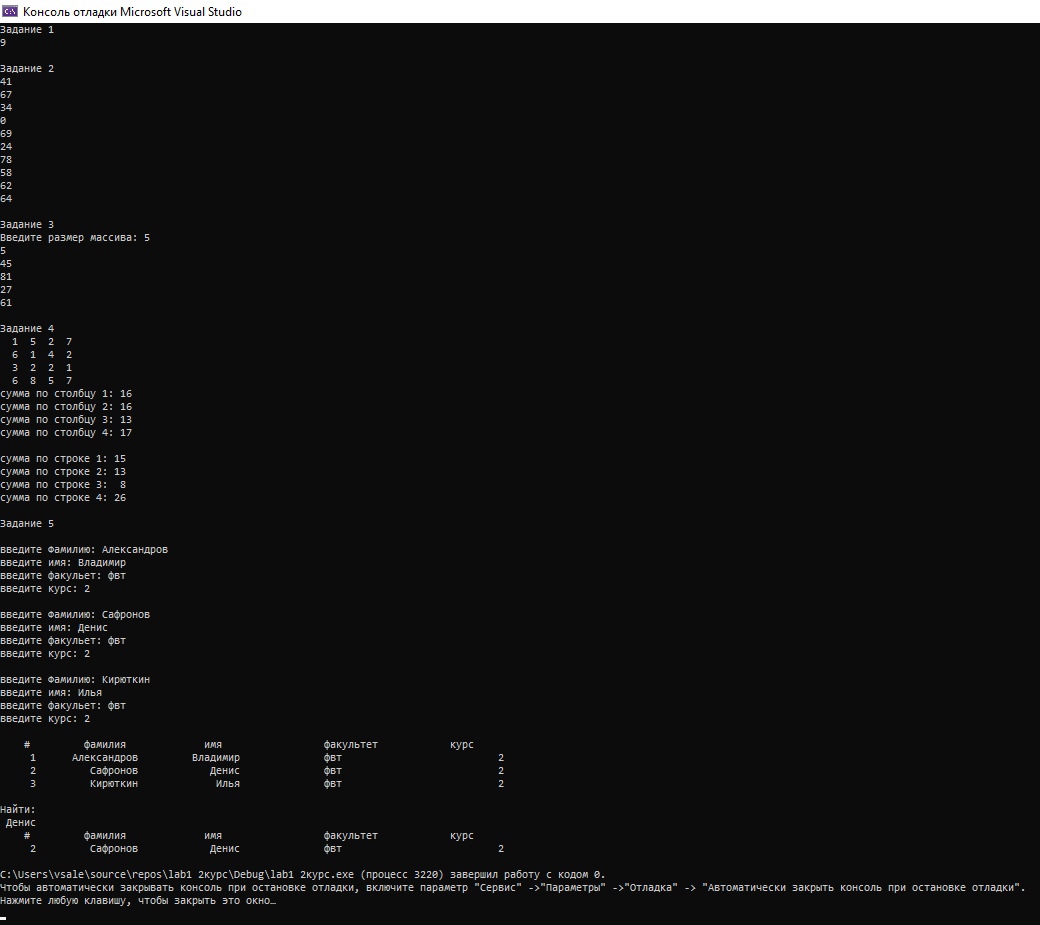








### Результаты работы программы



### Контрольный просчёт:

Вывод программы совпал с введёнными данными.

### Протокол трассировки программы

### 

На рисунке показан протокол трассировки после ввода всех значений

**Вывод:** Я вспомнил простые структуры данных, их реализацию, способы инициализации и заполнения массивов, а так же поиск в структурах.