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 19ВВ2

Кубасов И.М.

Принял: Д.Т.Н., профессор

Митрохин М.А.

Пенза 2020

**Цель работы:** получить навыки работы с простыми структурами данных

**Листинг:**

lab1.h

#pragma once

int diff(int\* arr, int length);

int random(int\* arr, int length, int seed);

int\* makearr(int size);

int summs(int\*\*& arr, int rows, int cols);

void eios(void);

lab1.cpp

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <conio.h>

#include <windows.h>

#include <time.h>

#include "lab1.h"

#define LENGTH 8

#define ROWS 3

#define COLS 4

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int a[] = { 0,1,2,3,4,5,6,7,8,9 };

int mas\_size;

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

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

}

printf("Разность между мин. и макс. элементом равна %d.\n", diff(a, 10));

int mas[LENGTH];

random(mas, LENGTH, time(0));

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

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

}

printf("\n");

printf("Введите размер создаваемого массива: ");

scanf("%d", &mas\_size);

printf("\n");

int\* dynarr = makearr(mas\_size);

random(dynarr, mas\_size, time(0) + 1);

for (int i = 0; i < mas\_size; i++) {

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

}

printf("\n\n");

int\*\* rectarr = (int\*\*)calloc(ROWS, sizeof(int\*));

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

rectarr[i] = (int\*)calloc(COLS, sizeof(int));

random(rectarr[i], COLS, time(0) + i + 2);

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

printf("%d\t", rectarr[i][j]);

}

printf("\n");

}

summs(rectarr, ROWS, COLS);

eios();

\_getch();

}

difference.cpp

int diff(int\* arr, int length) {

int min = arr[0];

int max = arr[0];

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

if (arr[i] < min)

min = arr[i];

if (arr[i] > max)

max = arr[i];

}

return (max-min);

}

random.cpp

int diff(int\* arr, int length) {

int min = arr[0];

int max = arr[0];

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

if (arr[i] < min)

min = arr[i];

if (arr[i] > max)

max = arr[i];

}

return (max-min);

}

size.cpp

#include <stdlib.h>

int\* makearr(int size) {

return (int \*) calloc(size, sizeof(int));

}

Summ.cpp

#include <stdio.h>

#include <stdlib.h>

int summs(int\*\*& arr, int rows, int cols) {

int\* summr = (int\*)calloc(rows, sizeof(int));

int\* summc = (int\*)calloc(cols, sizeof(int));

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

summc[i] = 0;

}

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

summr[i] = 0;

}

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

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

summr[i] += arr[i][j];

summc[j] += arr[i][j];

}

}

printf("Сумма по строкам: ");

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

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

}

printf("\n");

printf("Сумма по столбцам: ");

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

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

}

printf("\n");

return 0;

}

search.cpp

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <windows.h>

struct student

{

char famil[20];

char name[20], facult[20];

int Nomzach;

};

void eios(void) {

setvbuf(stdin, NULL, \_IONBF, 0);

setvbuf(stdout, NULL, \_IONBF, 0);

student stud[3];

int i;

char flag;

student compare;

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

{

printf("Введите фамилию студента\n"); scanf("%20s", stud[i].famil);

}

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

{

printf("Введите имя студента %s\n", stud[i].famil); scanf("%20s", stud[i].name);

}

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

{

printf("Введите название факультета студента %s %s\n", stud[i].famil, stud[i].name); scanf("%20s", stud[i].facult);

}

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

{

printf("Введите номер зачётной книжки студента %s %s\n", stud[i].famil, stud[i].name); scanf("%d", &stud[i].Nomzach);

}

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

{

printf("Cтудент %s %s обучается на факультете %s, номер зачётной книжки %d \n", stud[i].famil, stud[i].name,

stud[i].facult, stud[i].Nomzach);

}

printf("\nВведите фамилию искомого студента (если не важно, введите 0)\n"); scanf("%20s", compare.famil);

printf("Введите имя искомого студента (если не важно, введите 0)\n"); scanf("%20s", compare.name);

printf("Введите факультет искомого студента (если не важно, введите 0)\n"); scanf("%20s", compare.facult);

printf("Введите номер зачётной книжки искомого студента (если не важно, введите 0)\n"); scanf("%d", &compare.Nomzach);

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

flag = 1;

if (strcmp(compare.famil, "0")) {

if (strcmp(compare.famil, stud[i].famil)) {

flag = 0;

}

}

if (flag && strcmp(compare.name, "0")) {

if (strcmp(compare.name, stud[i].name)) {

flag = 0;

}

}

if (flag && strcmp(compare.facult, "0")) {

if (strcmp(compare.facult, stud[i].facult)) {

flag = 0;

}

}

if (flag && compare.Nomzach != 0) {

if (compare.Nomzach != stud[i].Nomzach) {

flag = 0;

}

}

if (flag) {

printf("Cтудент %s %s обучается на факультете %s, номер зачётной книжки %d \n", stud[i].famil, stud[i].name,

stud[i].facult, stud[i].Nomzach);

}

}

}

**Ход работы:**

**Задание 1**: написать программу, вычисляющую разницу между максимальным и минимальным элементами массива.

**Результат работы программы:**



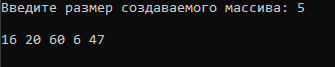
**Задание 2**: написать программу, реализующую инициализацию массива случайными числами.

**Результат работы программы:**

****

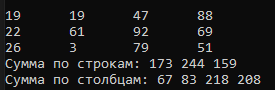
**Задание 3**: написать программу, реализующую создание массива произвольного размера, вводимого с клавиатуры.

**Результат работы программы:**

****

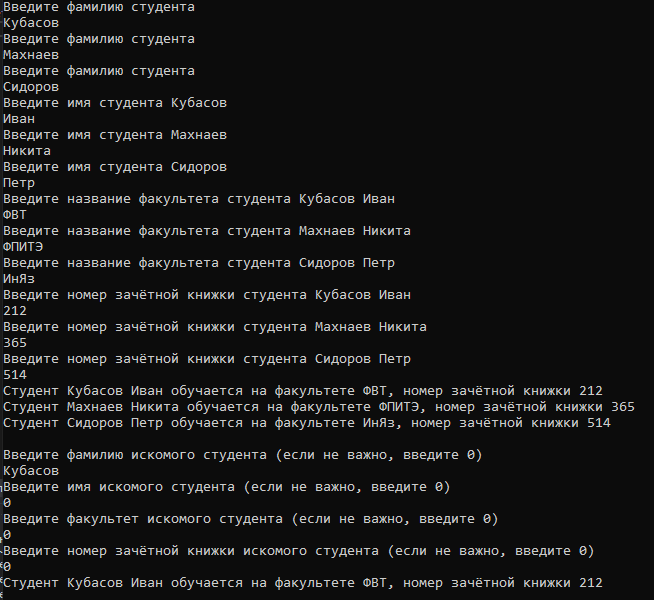
**Задание 4**: написать программу, вычисляющую сумму значений в каждом столбце (или строке) двумерного массива.

**Результат работы программы:**

****

**Задание 5**: написать программу, осуществляющую поиск среди структур student структуру с заданными параметрами (фамилией, именем и т.д.).

**Результат работы программы:**



**Вывод:** получил навыки работы с простыми структурами данных.