МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное образовательное учреждение высшего образования

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕХНОЛОГИЙ И УПРАВЛЕНИЯ

УНИВЕРСИТЕТСКИЙ КОЛЛЕДЖ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

ОТЧЕТ О ВЫПОЛНЕНИИ

ЛАБОРАТОРНОЙ РАБОТЫ №3

«Одномерные массивы»

Студента 2 курса, группы ИСП-207

Асылбек уулу Бакыта

Направление 09.02.07 – «Информационные системы и программирование»

Руководитель:

преподаватель

М.В. Валеев

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Работа защищена

«\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_»

“\_\_\_\_” \_\_\_\_\_\_\_\_\_\_\_\_\_2022г.

Москва, 2022

Оглавление

[Часть 1 3](#_Toc102810988)

[1. Постановка задачи 3](#_Toc102810989)

[2. Структурные требования 3](#_Toc102810990)

[3. Описание задачи 3](#_Toc102810991)

[3.1. Используемые переменные 3](#_Toc102810992)

[3.2. Псевдокод 3](#_Toc102810993)

[3.3. Блок-схема 5](#_Toc102810994)

[3.4. Листинг программной реализации 9](#_Toc102810995)

[3.5. Результаты 11](#_Toc102810996)

[Часть 2 12](#_Toc102810997)

[1. Постановка задачи 12](#_Toc102810998)

[2. Структурные требования 12](#_Toc102810999)

[3. Описание задач 12](#_Toc102811000)

[3.1. Используемые переменные 12](#_Toc102811001)

[3.2. Псевдокод 12](#_Toc102811002)

[3.3. Блок-схема 13](#_Toc102811003)

[3.4. Листинг программной реализации 17](#_Toc102811004)

[3.5. Результаты 19](#_Toc102811005)

[3.6. Тесты 19](#_Toc102811006)

[СПИСОК ЛИТЕРАТУРЫ 20](#_Toc102811007)

# Часть 1

## Постановка задачи

Обязательные для всех вариантов задания:

1. Найти максимальный и минимальный элементы массива

2. Переставить местами максимальный и минимальный элементы

3. Найти количество различных элементов в массиве. Вывести значения

неповторяющихся элементов.

## Структурные требования

## 3. Описание задачи

### 3.1. Используемые переменные

a – переменная, для определения неповторяющихся элементов;

b – переменная, для перестановки местами максимального и минимального элементов;

c – переменная, для хранения кол-ва неповторяющихся элементов;

n – кол-во элементов в массиве;

i – индекс;

j – переменная, для определения повторяющихся элементов;

imin – переменная, для определения минимального элемента;

imax – переменная, для определения максимального элемента;

max – максимальный элемент;

min – минимальный элемент;

M – массив.

### 3.2. Псевдокод

Начало

Ввод: n

Если вывод: введите число больше 0

Иначе для

Вывод:

Для

Если

Иначе если

Вывод:

Для

Вывод:

Для

Для

Если

Если

Вывод:

Вывод: c

Конец

### 3.3. Блок-схема

Начало

Ввод: n

Вывод:

M[i]<min

Вывод:

Вывод: не повторяющиеся элементы

Вывод:

Вывод: Неповторяющиеся элементы

Вывод:

Вывод: c

Конец

вывод: введите число больше 0

### 3.4. Листинг программной реализации

#include <stdio.h>

#include <time.h>

#include <stdlib.h>

int main()

{

srand(time(NULL));

int\* M, a, b, c = 0, n, imin = 0, imax = 0;

printf(" Введите кол-во элементов: \n");

scanf("%i", &n);

if (n < 1) printf("Введите число больше 0\n");

else

{

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

{

M[i] = ((RAND\_MAX) / rand());

printf(" %i", M[i]);

}

int max = M[0], min = M[0];

for (int i = 1; i < n; i++)

{

if (M[i] > max)

{

max = M[i];

imax = i;

}

if (M[i] < min)

{

min = M[i];

imin = i;

}

}

printf("\n максимальный элемент %i\n минимальный элемент %i", M[imax], M[imin]);

b = M[imax];

M[imax] = M[imin];

M[imin] = b;

printf("\n Перестановка мест максимального и минимального элементов\n");

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

printf(" %i", M[i]);

printf("\n Неповторяющиеся элементы: ");

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

{

a = 1;

for (int j = 0; j < n; j++)

{

if ((M[i] == M[j]) && (i != j))

{

a = 0;

}

}

if (a == 1)

{

printf(" %i", M[i]);

c++;

}

}

printf("\n Кол-во различных элементов в массиве %i", c);

}

return 0;

}

### 3.5. Результаты

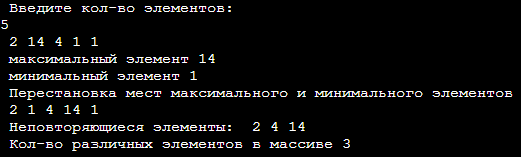


Рисунок – 1



Рисунок – 2

# Часть 2

## Постановка задачи

Вариант 1

В одномерном массиве, состоящем из п вещественных элементов, вычислить:

1) сумму отрицательных элементов массива;

2) произведение элементов массива, расположенных между максимальным и минимальным элементами.

## Структурные требования

## Описание задач

### Используемые переменные

n – размерность массива;

s – переменная, для вычисления суммы отрицательных элементов массива;

z – переменная, для вычисления произведения;

imin – переменная, для определения минимального элемента;

imax – переменная, для определения максимального элемента;

max – максимальный элемент;

min – минимальный элемент;

### Псевдокод

Начало

Ввод: n

Если вывод: Попробуй снова

Иначе для

Вывод:

Для

Если

Если

Вывод:

Для

Если

Вывод: s

Если

Для i = imin + 1, i < imax

Иначе если

Для

Вывод: z

Конец

### Блок-схема

Начало

Ввод: n

Вывод:

M[i]<min

Вывод:

Вывод: s

imax > imin

i = imin + 1,

i < imax

imin > imax

i = imax + 1,

i < imin

Вывод: z

Конец

Вывод: попробуй снова

### Листинг программной реализации

#include <stdio.h>

#include <time.h>

#include <stdlib.h>

int main() {  
 srand(time(NULL));  
 int n;  
 printf("Enter the number of elements: \n");  
 scanf("%d", &n);  
 if (n <= 0) return printf("error");  
 float array[n], z = 1;  
 for (int i = 0; i < n; i++) {  
 array[i] = (z \* (float)(RAND\_MAX) / (float)rand());  
 z = -z;  
 printf(" %.2f", array[i]);  
 }  
 int iMin = 0, iMax = 0;  
 float s = 0, max = array[0], min = array[0];  
 for (int i = 0; i < n; i++) {  
 if (array[i] > max) {  
 max = array[i];  
 iMax = i;  
 }  
 if (array[i] < min) {  
 min = array[i];  
 iMin = i;  
 }  
 }  
 printf("\n max element = %.1f\n min element = %.1f", array[iMax], array[iMin]);  
 for (int i = 0; i < n; i++) {  
 if (array[i] < 0) s += array[i];  
 }  
 printf("\n sum of negative elements = %.1f", s);  
 if (iMax > iMin) {  
 for (int i = iMin; i < iMax; i++) {  
 z \*= array[i];  
 }  
 } else {  
 for (int i = iMax; i < iMin; i++) {  
 z \*= array[i];  
 }  
 }  
 printf("\n multiplication of elements between max and min = %.1f\n", z);  
 return 0;  
}

### Результаты

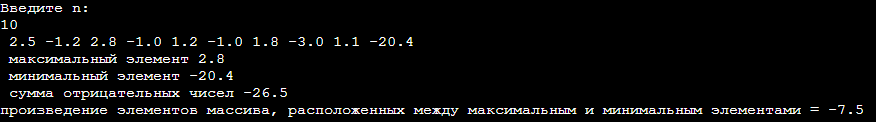


Рисунок – 3



Рисунок – 4

### Тесты

|  |  |
| --- | --- |
| Ручной | Программный |
| -26,6 | -26,5 |
| -7,128 | -7,5 |

# СПИСОК ЛИТЕРАТУРЫ

1. Конова Е.А., Поллак Г.А. Алгоритмы и программы. Язык С++: Учебное

пособие. - 2-е изд., стер. - СПб.: Издательство "Лань", 2017. - 384 с.

2. Седжвик Роберт. Алгоритмы на С++.: Пер. с англ. - М.: ООО "И.Д. Вильямс",

2011. - 1056 с.

3. Лафоре Р. Объектно-ориентированное программирование в С++. Классика

Computer Science. 4-е изд. - СПб.: Питер, 2015. - 928 с.

4. Орлов С.А. Теория и практика языков программирования: Учебник для вузов.

Стандарт 3-го поколения. - СПб.: Питер, 2014. - 688 с.

5. Павловская Т.А. С/С++. Процедурное и объектно-ориентированное

программирование: Учебник для вузов. Стандарт 3-го поколения. - СПб.:

Питер, 2015. - 496 с.

6. Павловская Т.А. С#. Программирование на языке высокого уровня: Учебник

для вузов. - СПб.: Питер, 2014. - 432 с.

7. Плаксин М.А. Тестирование и отладка программ - для профессионалов

будущих и настоящих. - М.: БИНОМ. Лаборатория знаний, 2013. - 167 с.