

Федеральное агентство морского и речного транспорта

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Государственный университет морского и речного флота имени адмирала С.О. Макарова»

Институт водного транспорта

Кафедра вычислительных систем и информатики

**ОТЧЕТ**

**По лабораторной работе на тему**

**«Лабораторная работа №1»**

**Дисциплина: Технологии и методы программирования**

**Вариант №1**

|  |  |
| --- | --- |
|  | **Выполнил**: студент гр. ИC-21  Донской Д.А.  **Руководитель**: доц. каф. ВСИ  к.г.н. Балса А. Р. |
|  |  |

Санкт-Петербург

2025

**ПОСТАНОВКА ЗАДАЧИ 1**

В одномерном массиве, состоящем из n вещественных элементов:

1. вычислить сумму отрицательных элементов массива;
2. вычислить произведение элементов массива, расположенных между максимальным и минимальным элементом;
3. упорядочить элементы массива по возрастанию. (Quick sort.)

**АНАЛИЗ**

**Задача 1.** вычислить сумму отрицательных элементов массива

**Не успешные сценарии:**

1. Проверка размера массива

**ПОСМОТРЕТЬ ПРИЛОЖЕНИЕ В КОНЦЕ ДОКУМЕНТА!**

**Ввод**: размер массива.

**Вывод:**

**Ответ**:

1. **Отсутствие отрицательных элементов в массиве**

Пусть в массиве 3 элемента

**Ввод: 1 0 3**

**Вывод**: Как видим, элементов отрицательных нет.

**Ответ**: **Задача не может быть выполнена!**

Объяснение пользователю, что в массиве нет отрицательных элементов.

1. **В массиве только один отрицательный элемент**

Пусть в массиве 3 элемента

**Ввод: 1 0 -4**

**Вывод**: Как видим, отрицательный элемент один и сумму найти не удастся.

**Ответ**: **Задача не может быть выполнена!**

Объяснение пользователю, отрицательный элемент один.

**Задача 2.** вычислить произведение элементов массива, расположенных между максимальным и минимальным элементом

**Не успешные сценарии:**

1. **Проверка размера массива**
2. **Максимальный и минимальный элементы расположены друг за**

**другом**

Пусть в массиве 3 элемента

**Ввод: 1 8 5**

**Вывод**: Как видим, максимальный и минимальный элемент стоят подряд

**Ответ**: объяснение пользователю, что в массиве нельзя подсчитать произведение элементов, расположенных между максимальным и минимальным элементами, так как максимальный и минимальный стоят подряд

1. **Присутствие несколько максимумов и минимумов** Пусть в массиве 4 элемента:

**Ввод: 3 3 3 3**

**Вывод**: Как мы видим, все элементы являются максимумами и минимумами.

Дадим пользователю выбрать позиции минимального и максимального элементов.

**Рассмотрим следующие возможные сценарии:**

* 1. Пользователь выбирает одинаковые позиции максимума и минимума.

**Ответ:** Сумма не может быть найдена, вы выбрали одинаковые позиции для максимума и минимума.

* 1. Пользователь выбрал позиции максимума и минимума элементов друг за другом.

**Ответ:** Сумма не может быть найдена, вы выбрали позиции максимума и минимума элементов друг за другом.

* 1. Пользователь выбрал позиции максимума и минимума элементов через одну позицию.

**Ответ:** Сумма не может быть найдена, вы выбрали позицию максимума и минимума через один элемент.

Все остальные сценарии будут являться успешными.

* 1. **Задача 3.** упорядочить элементы массива по возрастанию. (Quick sort.)

**Вывод:** любой массив может быть отсортирован.

**ПОСТАНОВКА ЗАДАЧИ 2**

Дана целочисленная прямоугольная матрица. Определить:

1) количество строк, не содержащих ни одного нулевого элемента ; 2) максимальное из чисел, встречающихся в заданной матрице более одного раза.

**АНАЛИЗ**

**Задача 1.** Определить количество строк, не содержащих ни одного

нулевого элемента **Не успешные сценарии:**

1. **Проверка размеров массива**

**ПОСМОТРЕТЬ ПРИЛОЖЕНИЕ В КОНЦЕ ДОКУМЕНТА!**

1. **Отсутствие нулевых элементов в массиве** Пусть в матрице 4 элемента:

**Ввод:**  0 2

3 0

**Ответ:** Объяснение пользователю, что нет строк без нулевых элементов.

**Задача 2.** Определитьмаксимальное из чисел, встречающихся в заданной матрице более одного раза **Не успешные сценарии:**

1. **Проверка размера матрицы.**
2. **Отсутствие одинаковых элементов в матрице.**

Пусть в матрице 4 элемента :

**Ввод:** 1 2

3 4

**Ответ:** Объяснение пользователю, что нет в матрице одинаковых элементов.

**Успешные сценарии:**

Пусть в матрице 4 элемента :

**Ввод:** 1 2

2 1

**Ответ:** Максимальное значение, встречающееся более одного раза - 2

Позиция [0;0].

**ПОСТАНОВКА ЗАДАЧИ 3**

Написать программу, которая считывает из текстового файла три предложения и выводит их в обратном порядке.

**АНАЛИЗ**

**Неуспешные сценарии:**

1. **проверка открытия файла**

Файл не существует

**Ответ**: Объяснение пользователю, что в файл не удается открыть.

1. **проверка наличия текста в файле** Пусть в файле нет текста.

**Ответ**: Объяснение пользователю, что файл пуст.

1. **В файле меньше трех предложений**.

Пусть в файле запись:

*Предложение первое. Предложение второе!*

Как видим, файл в таком случае состоит из двух предложенийй.

**Ответ:** Объяснение пользователю, что в файле меньше трех предложений.

**ПРИЛОЖЕНИЕ**

**Пример проверки размера одномерного массива**

**#include <iostream>**

**using namespace std;**

**const int MAX\_BUF = 200; // Максимальный размер массива**

**int main() {**

**int n = 0;**

**char Ok;**

**int\* A = new int[MAX\_BUF];**

**cout << "Please, enter the size of your array not bigger than " << MAX\_BUF << " -> ";**

**cin >> n;**

**while (true) {**

**if (n <= 0 || n > MAX\_BUF) {**

**cout << "It's so sorry! The size is not correct!\n"**

**<< "If you wish to quit, click 'Y' or any other key to continue -> ";**

**cin >> Ok;**

**if (Ok == 'y' || Ok == 'Y') {**

**cout << "Good bye!";**

**break;**

**} else {**

**cout << "Please, enter again the size of your array not bigger than "**

**<< MAX\_BUF << " -> ";**

**cin >> n;**

**}**

**} else {**

**break;**

**}**

**}**

**// Ввод элементов массива**

**cout << "Enter the array's elements:\n";**

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

**cin >> A[i];**

**}**

**// Вывод элементов массива**

**cout << "You have entered the next elements:\n";**

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

**cout << A[i] << " ";**

**}**

**cout << "\n\n\*\*\*\*\*\* For the moment it's all! Bye! \*\*\*\*\*\*\*\n\n";**

**delete[] A;**

**A = nullptr;**

**return 0;**

}