**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра Вычислительной техники**

отчет

**по лабораторной работе № 2**

**по дисциплине «Программирование»**

Тема: Обработка одномерных массивов

|  |  |  |
| --- | --- | --- |
| Студент гр. 3312 |  | Шарапов И. Д. |
| Преподаватель |  | Аббас С. А. |

Санкт-Петербург

2023

**Содержание**

[Цель работы 3](#_Toc146535796)

[Задание (Вариант 14) 3](#_Toc146535797)

[Постановка задачи и описание решения 3](#_Toc146535798)

[Описание переменных 3](#_Toc146535799)

[Схема алгоритма 4](#_Toc146535800)

[Текст программы 5](#_Toc146535801)

[Контрольные примеры 5](#_Toc146535802)

[Примеры выполнения программы 6](#_Toc146535803)

[Выводы 7](#_Toc146535804)

# Цель работы

Целью работы является изучение особенностей работы с массивами в языке Си и получение практических навыков в решение задач, в которых необходимо обрабатывать множество значений.

# Задание (Вариант 14)

Определить, есть ли в массиве хотя бы одна тройка соседних чисел, в которой средний элемент больше своих «соседей», т. е. предшествующего и последующего. В случае положительного ответа определить номера элементов первой из таких троек.

# Постановка задачи и описание решения

Для решения задачи необходимо считать и запомнить элементы массива, который ввёл пользователь. Сделаем это с помощью цикла “for”. Будем хранить ответ на задачу в переменной ans. Изначально ans будет равно -1, так как этого индекса не может быть в реальном ответе, и, если ans так и останется отрицательным, значит мы не нашли искомую тройку. После этого переберём тройки соседних чисел. Их количество n – 2, где n – длина массива.

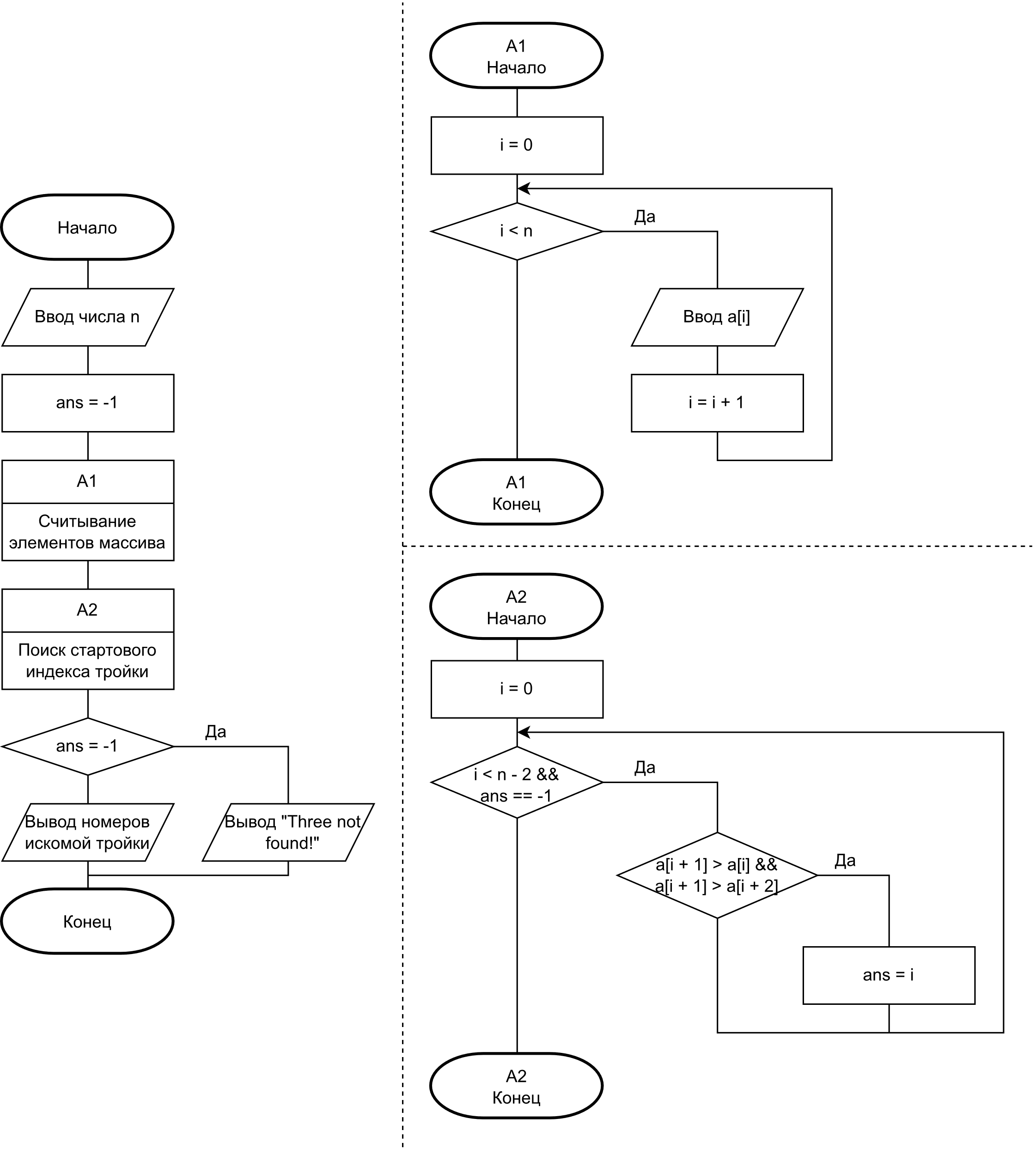
Рассмотрим каждую конкретную тройку по очереди. Если мы встречаем удовлетворяющую условию тройку, мы записываем индекс первого её элемента в переменную ans и завершаем перебор.

Если ans не изменилось, т.е. осталось равным -1, искомой тройки нет, иначе выведем на экран порядковые номера элементов этой тройки в массиве.

# Описание переменных

|  |  |  |  |
| --- | --- | --- | --- |
| № | Имя переменной | Тип | Назначение |
| 1 | n | int | Количество элементов в массиве |
| 2 | ans | int | Индекс первого элемента в искомой тройке |
| 3 | a[n] | int[] | Исходный массив |

# Схема алгоритма



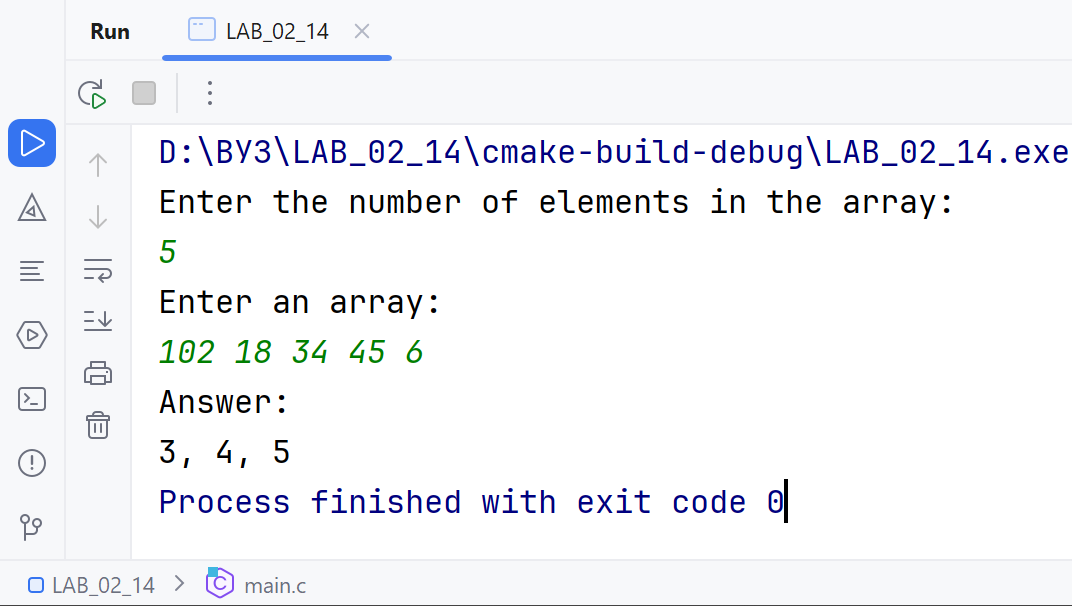
# Текст программы

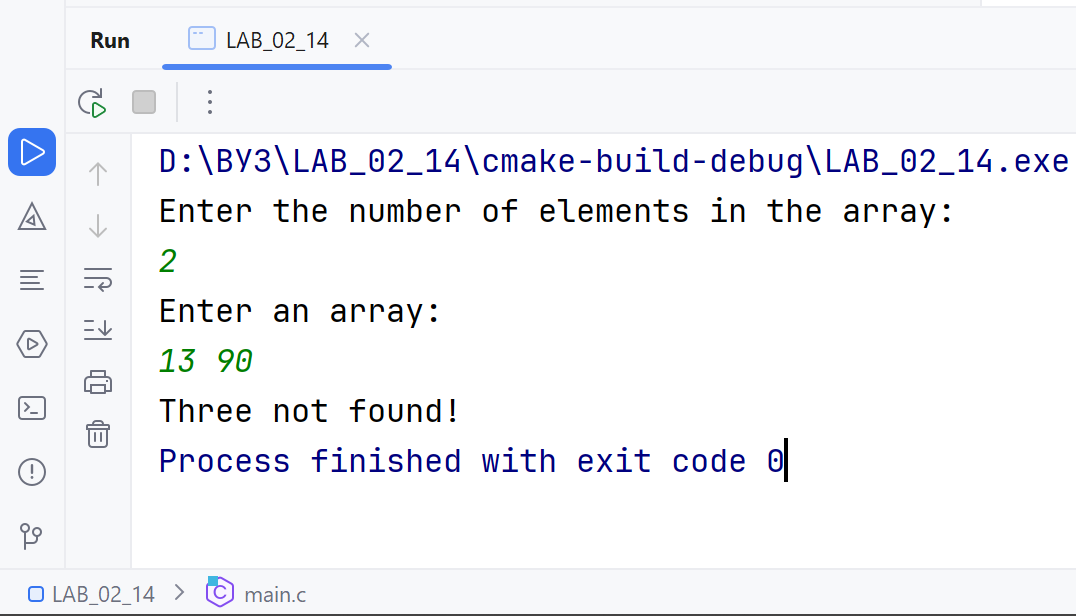
|  |
| --- |
| #include <stdio.h>  int main() {  printf("Enter the number of elements in the array:\n");  int n, ans;  scanf("%i", &n);  printf("Enter an array:\n");  int a[n];  ans = -1;  for (int i = 0; i < n; ++i) {  scanf("%i", &a[i]);  }   for (int i = 0; i < n - 2 && ans == -1; ++i) {  if (a[i + 1] > a[i] && a[i + 1] > a[i + 2]) {  ans = i;  }  }  if (ans == -1) {  printf("Three not found!");  } else {  printf("Answer:\n");  printf("%i, %i, %i", ans + 1, ans + 2, ans + 3);  }   return 0; } |

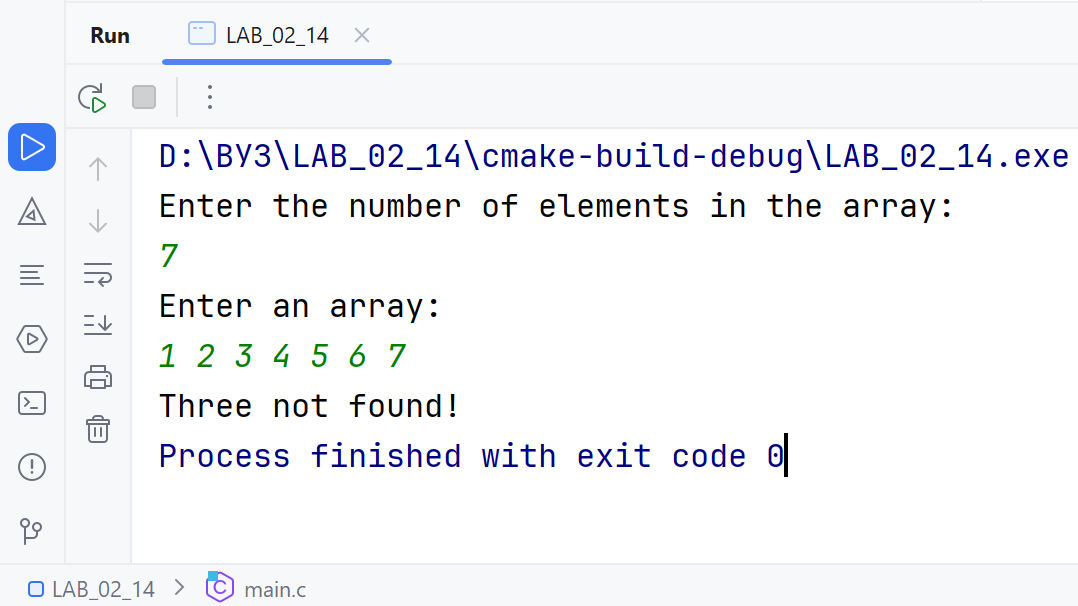
# Контрольные примеры

|  |  |  |
| --- | --- | --- |
| № | Исходные данные | Результаты |
| 1 | 5  102 18 34 45 6 | 3, 4, 5 |
| 2 | 2  13 90 | Three not found! |
| 3 | 7  1 2 3 4 5 6 7 | Three not found! |

# Примеры выполнения программы







# Выводы

В результате выполнения работы изучены особенности массива в языке Си. А также получены практические навыки в решении задач, где требуется обработать неизвестное заранее количество чисел.