Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

 Лабораторная работа 10

По дисциплине «Основы алгоритмизации и программирования»

На тему «Одномерные массивы»

Выполнил:

Студент 1 курса 10 группы

Сегренёв Кирилл Сергеевич

Преподаватель: асс. Андронова М.В.

2023, Минск

**ВАРИАНТ 14**

В соответствии со своим вариантом написать программы по условиям, представленным в таблице ниже. Использовать одномерный массив целых случайных чисел (диапазон от 0 до 99). Размер массива ввести с клавиатуры.

Представить результаты в окне **Отладчика**.

Для одной из программ составить ***блок-схему*** *а*лгоритма.

**Задание 1.**

В массиве несколько нулевых элементов. Найти первый и последний нулевые элементы. Вывести их индексы.

**Код:**

#include <iostream> // Подключение библиотеки для работы с вводом/выводом.

#include <vector> // Подключение библиотеки для работы с массивом.

using namespace std; // Использование пространства имен std

void main()

{

setlocale(LC\_CTYPE, "Russian"); // Подключаем функцию, с помощью которой корректно отображается русский язык

int a; // Объявление переменной a

cout << "Введите длину массива: "; // Вывод на экран предложения ввести длину массива

cin >> a; // Считываем с клавиатуры символ a

vector<int> array(a); // Объявляем динамический массив с длинной, указанной пользорвателем

int zerofirst = -1, zerolast = -1; // Объявление переменных типа int. Применяем такие значения для того, чтобы в дальнейшем находить нулевые значения и присваивать им индекс

cout << "Введите числа последовательности: "; // Вывод на экран предложения ввести последовательность чисел

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

cin >> array[n]; // Ввод элемента последовательности

if (array[n] == 0) { // Сравнение этого элемента с заданными значениями

if (zerofirst == -1) {

zerofirst = n;

}

zerolast = n;

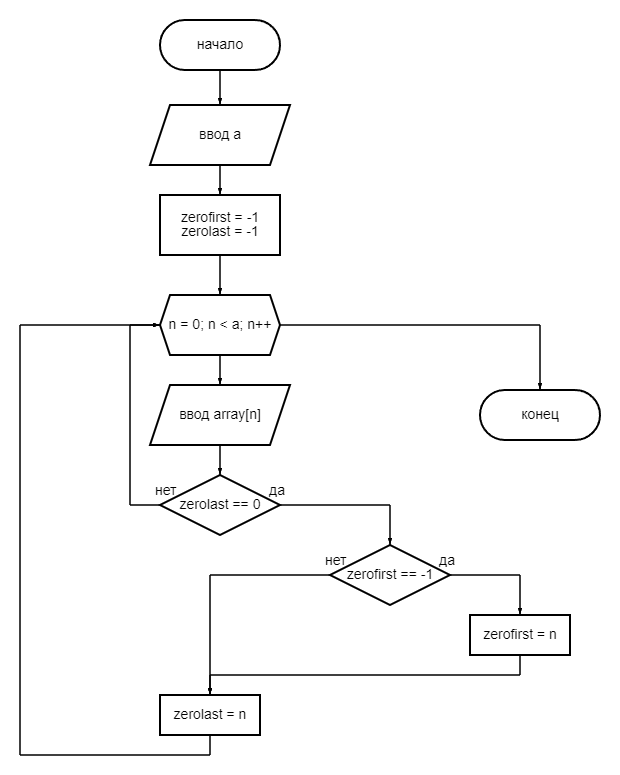
}

}

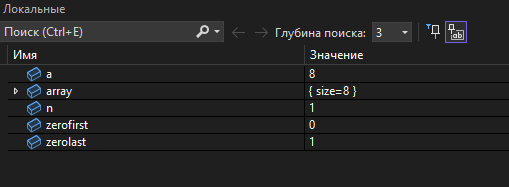
cout << "Индекс первого нулевого элемента: " << zerofirst << '\n' << "Индекс второго нулевого элемента: " << zerolast; // Вывод индексов нулевых элементов

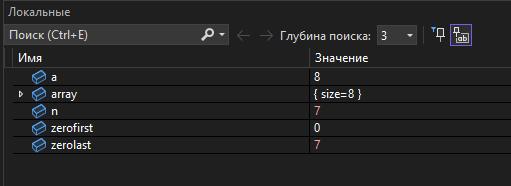
}

**Блок-схема:**

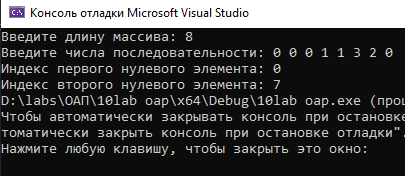
****

**Отладчик:**

****

****

**Вывод:**

****

**Задание 2.**

Задан массив из k чисел. Найти число, наиболее часто встречающееся в этом массиве.

**Код:**

#include <iostream> // Подключение библиотеки для работы с вводом/выводом.

#include <vector> // Подключение библиотеки для работы с массивом.

using namespace std; // Использование пространства имен std

int main() {

setlocale(LC\_ALL, "rus"); // Подключаем функцию, с помощью которой корректно отображается русский язык

int k, counter = 0; // Объявление переменной k и counter

cout << "Введите длину массива: "; // Вывод на экран предложения ввести длину массива

cin >> k; // Считываем с клавиатуры символ k

vector<int> array(k); // Объявляем динамический массив с длинной, указанной пользорвателем

cout << "Введите числа последовательности: "; // Вывод на экран предложения ввести последовательность чисел

for (int n = 0; n < k; n++) { // Ввод последовательности чисел

cin >> array[n];

}

for (int n = 0; n < k; n++) { // Использование цикла для того, чтобы сравнивать рядом стоящие элементы

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

if (array[n] == array[j]) {

counter = 1;

}

if (array[n] != array[j]) {

counter = 0;

break;

}

}

if (counter == 0) {

cout << "Число " << array[n] << " повторяется чаще всего." << endl; // Вывод элемента массива, который повторяется чаще всего

break;

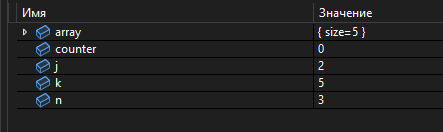
}

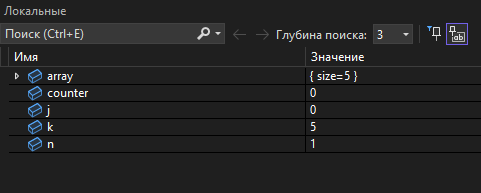
}

return 0;

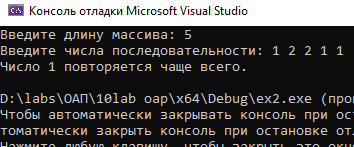
}

**Отладчик:**

****

****

**Вывод:**

****

**ДОПОЛНИТЕЛЬНЫЕ ЗАДАНИЯ**

**Задание 1.**Имеются результаты **n** ежедневных измерений количества выпавших осадков. За какую из недель (отрезок времени длиной 7 дней), считая с начала периода измерений, выпало наибольшее количество осадков?

**Код:**

#include <iostream> // Подключение библиотеки для работы с вводом/выводом.

using namespace std; // Использование пространства имен std

int main() {

setlocale(LC\_ALL, "rus"); // Подключаем функцию, с помощью которой корректно отображается русский язык

const int n = 7; // Инициализируем размер массива

int k, array[n], sum = 0, max = 0, mark; // Объявляем и инициализируем переменные

cout << "Введите период измерений(в неделях): "; // Предложение польхователю ввести недели

cin >> k; // Считываем с клавиатуры символ k

for (int g = 0; g < k; g++) // Выполнение цикла для количества недель, веденных пользователем

{

cout << g + 1 << " Неделя: " << '\n'; // Вывод на экран число недели

sum = 0; // Инициализация переменной

for (int i = 0; i < n; i++) { // Цикл для нахождение максимального числа осадков для недели

cout << "Введите осадки за " << i + 1 << " день: ";

cin >> array[i];

sum += array[i];

}

if(max < sum){

max = sum;

mark = g + 1;

}

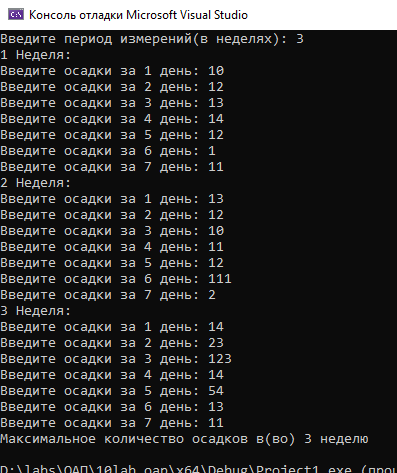
}

cout << "Максимальное количество осадков в(во) " << mark << " неделю" << '\n'; // Вывод на экран недели, у которой больше всего осадков

return 0;

}

**Вывод:**

****

**Задание 2.**

Подсчитать количество пар соседних элементов массива с одинаковыми значениями.

**Код:**

#include <iostream> // Подключение библиотеки для работы с вводом/выводом.

#include <vector> // Подключение библиотеки для работы с массивом.

using namespace std; // Использование пространства имен std

int main() {

setlocale(LC\_ALL, "rus"); // Подключаем функцию, с помощью которой корректно отображается русский язык

int k, g = 0; // Объявление переменной k

cout << "Введите длину массива: "; // Вывод на экран предложения ввести длину массива

cin >> k; // Считываем с клавиатуры символ k

vector<int> array(k); // Объявляем динамический массив с длинной, указанной пользорвателем

cout << "Введите числа последовательности: "; // Вывод на экран предложения ввести последовательность чисел

// Ввод каждого элемента массива

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

cin >> array[n];

}

int j = 0; // Инициализируем переменную

for (int n = 0; n < k; n++) { // Цикл для подсчета количества пар рядом стоящих одинаковых элементов

for (; j < n; j++) {

if (array[n] == array[j]) {

g++;

}

}

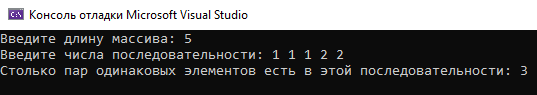
}

cout << "Столько пар одинаковых элементов есть в этой последовательности: " << g << endl; // Вывод на экран количество пар элементов

return 0;

}

**Вывод:**

****

**Задание 3.**

Найти в массиве наибольшее число подряд идущих одинаковых элементов.

**Код:**

#include <iostream> // Подключение библиотеки для работы с вводом/выводом.

#include <vector> // Подключение библиотеки для работы с массивом.

using namespace std; // Использование пространства имен std

int main() {

setlocale(LC\_ALL, "rus"); // Подключаем функцию, с помощью которой корректно отображается русский язык

int M, k, sum = 1, max = 0; // Объявляем и инициализируем переменные

cout << "Введите длину массива: "; // Вывод на экран предложения ввести длину массива

cin >> M; // Считываем с клавиатуры символ M

vector<int> array(M); // Объявляем динамический массив с длинной, указанной пользорвателем

cout << "Введите числа последовательности: "; // Вывод на экран предложения ввести последовательность чисел

// Ввод элементов последовательности

for (int g = 0; g < M; g++) {

cin >> array[g];

}

int i = 0; // Инициализация переменной

for (int g = 0; g < M; g++) { // Цикл для сравнения рядом стоящих элементов в массиве

for (; i < g; i++) {

if (array[g] == array[i]) {

sum++;

}

if (array[g] != array[i]) {

sum = 1;

}

if (max < sum) {

max = sum;

}

}

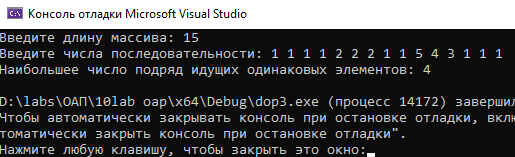
}

cout << "Наибольшее число подряд идущих одинаковых элементов: " << max << endl;

return 0;

}

**Вывод:**

****