МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт математики и информационных систем

Факультет автоматики и вычислительной техники

Кафедра систем автоматизации управления

Дата сдачи на проверку:

«\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2024 г.

Проверено:

«\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2024 г.

**«Разветвляющиеся алгоритмы»**

Отчет по самостоятельной работе № 2  
по дисциплине

«Информатика»

Вариант 5

Выполнил студент гр. ИТб-1302-02-20 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Ердяков Р.А./

(Подпись)

Проверил ст. преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Симонов А.Г./

(Подпись)

Киров 2024

# 1 ЦЕЛЬ РАБОТЫ

# Приобретение навыков организации ветвлений в алгоритмах.

# 2 ЗАДАНИЕ

Разработать два алгоритма определения количества неположительных чисел, среди заданных целых чисел a, b и c.

**3 СХЕМА АЛГОРИТМА**

Схема первого способа представлена на рисунке 1.

Рисунок 1

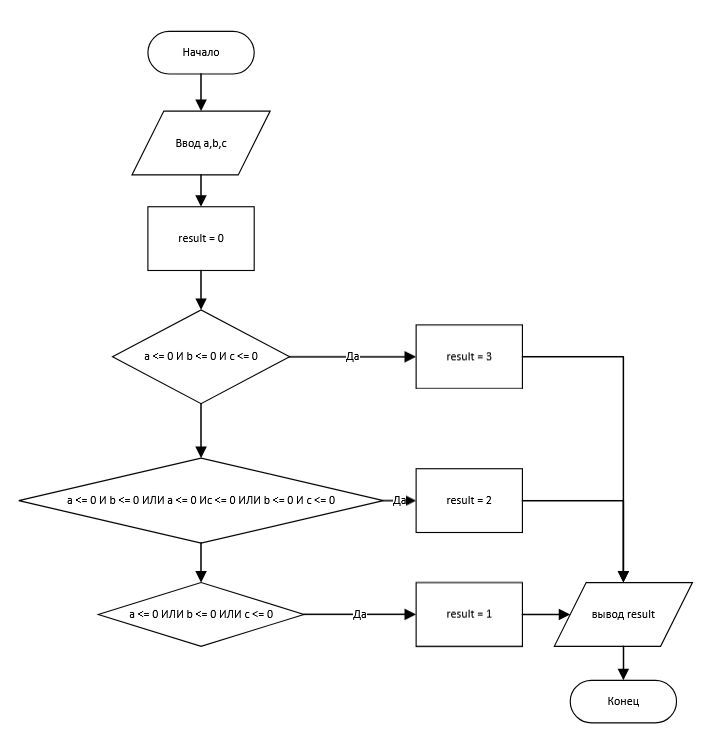
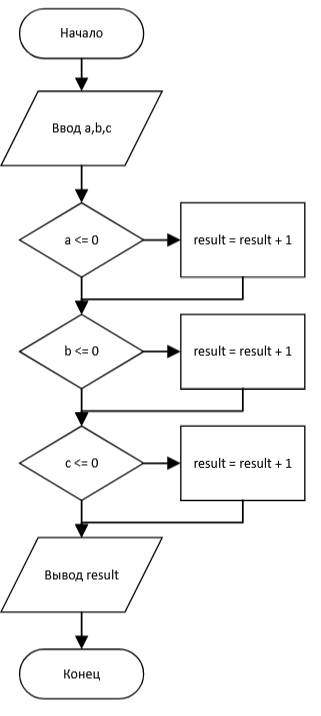


Схема второго способа представлена на рисунке 2.

Рисунок 2



**5 СПИСОК ИДЕНТИФИКАТОРОВ ПЕРЕМЕННЫХ И КОНСТАНТ**

Идентификаторы программы представлены в таблице 1.

Таблица 1 –- Идентификаторы программы и их тип

|  |  |  |
| --- | --- | --- |
| Назначение | Идентификатор | Тип |
| Вводимая переменная | a | int |
| Вводимая переменная | b | int |
| Вводимая переменная | c | int |
| Вычисляемая переменная, количество неположительных чисел | result | int |

**6 РЕЗУЛЬТАТЫ ТЕСТИРОВАНИЯ ПРОГРАММЫ.**

Результаты тестирования программы представлены в таблице 2.

Таблица 2 - Результаты тестирования программы

|  |  |  |  |
| --- | --- | --- | --- |
| Исходные данные | Ручной счет | Программный результат | Номер рисунка |
| a = 0  b = 0  c = 0 | 3 | 3 | Рисунок 3 |
| a = -2  b = 0  c = 1 | 2 | 2 | Рисунок 4 |
| a = 1  b = 10  c = 12 | 0 | 0 | Рисунок 5 |

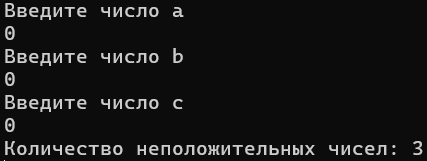


Рисунок 3 – Экранная форма программы с исходными данными

a = 0 b = 0 c = 0

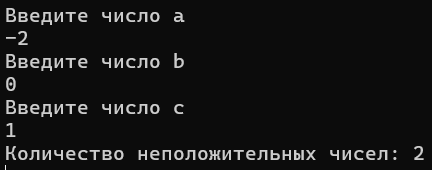


Рисунок 4 – Экранная форма программы с исходными данными

a = -2 b = 0 c = 1

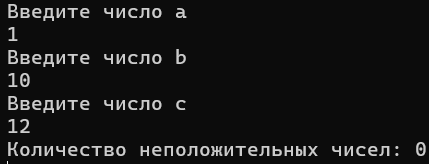


Рисунок 5 – Экранная форма программы с исходными данными

a = 1 b = 10 c = 12

**7 ВЫВОД.**

Приобретение навыков организации ветвлений в алгоритмах. Произведено сравнение двух способов проверки неположительных чисел. Способ 2 оказался эффективнее из-за меньшего количества логических операций 3 против 21 в первом способе.

**Приложение А**

**Листинг программы первого способа**

using System;

internal class Program

{

static void Main(string[] args)

{

Console.WriteLine("Введите число a");

int a = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Введите число b");

int b = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Введите число c");

int c = Convert.ToInt32(Console.ReadLine());

int result = 0;

if (a <= 0 && b <= 0 && c <= 0)

{

result = 3;

}

else if ((a <= 0 && b <= 0) || (a <= 0 && c <= 0) || (b <= 0 && c <= 0))

{

result = 2;

} else if (a <= 0 || b <= 0 || c <= 0)

{

result = 1;

}

Console.WriteLine("Количество неположительных чисел: " + result);

Console.ReadKey();

}

}

**Листинг программы первого способа**

using System;

internal class Program

{

static void Main(string[] args)

{

Console.WriteLine("Введите число a");

int a = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Введите число b");

int b = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Введите число c");

int c = Convert.ToInt32(Console.ReadLine());

int result = 0;

if (a <= 0)

{

result++;

}

if (b <= 0)

{

result++;

}

if (c <= 0)

{

result++;

}

Console.WriteLine("Количество неположительных чисел: " + result);

Console.ReadKey();

}

}