Пермский филиал федерального государственного автономного образовательного учреждения высшего образования «Национальный исследовательский университет «Высшая школа экономики»

*Факультет экономики, менеджмента и бизнес-информатики*

Шульжик Кирилл Владимирович

**Отчёт по практической работе №2 (Вариант 16)**

студента образовательной программы бакалавриата «Программная инженерия» по направлению подготовки 09.03.04 Программная инженерия

Преподаватель

В.Д Марквирер

Оглавление

[**Задача 1** 3](#_Toc85384721)

[**Анализ** 3](#_Toc85384722)

[**Алгоритм** 5](#_Toc85384723)

[**Программа** 6](#_Toc85384724)

[**Тесты** 7](#_Toc85384725)

[**Задача 2** 9](#_Toc85384726)

[**Анализ** 9](#_Toc85384727)

[**Алгоритм** 10](#_Toc85384728)

[**Программа** 11](#_Toc85384729)

[**Тесты** 12](#_Toc85384730)

[**Задача 3** 13](#_Toc85384731)

[**Анализ** 13](#_Toc85384732)

[**Алгоритм** 14](#_Toc85384733)

[**Программа** 15](#_Toc85384734)

[**Тесты** 16](#_Toc85384735)

**Задача 1**

Дана последовательность из n целых чисел. Найти количество элементов этой последовательности, кратных числу K1 и не кратных числу K2.

**Анализ**

Исходные данные:

– n – количество чисел в последовательности;

– k1, k2 целые числа;

– a1,a2…an – элементы последовательности.

Результат:

– Количество чисел, кратных K1 и не кратных K2 - count.

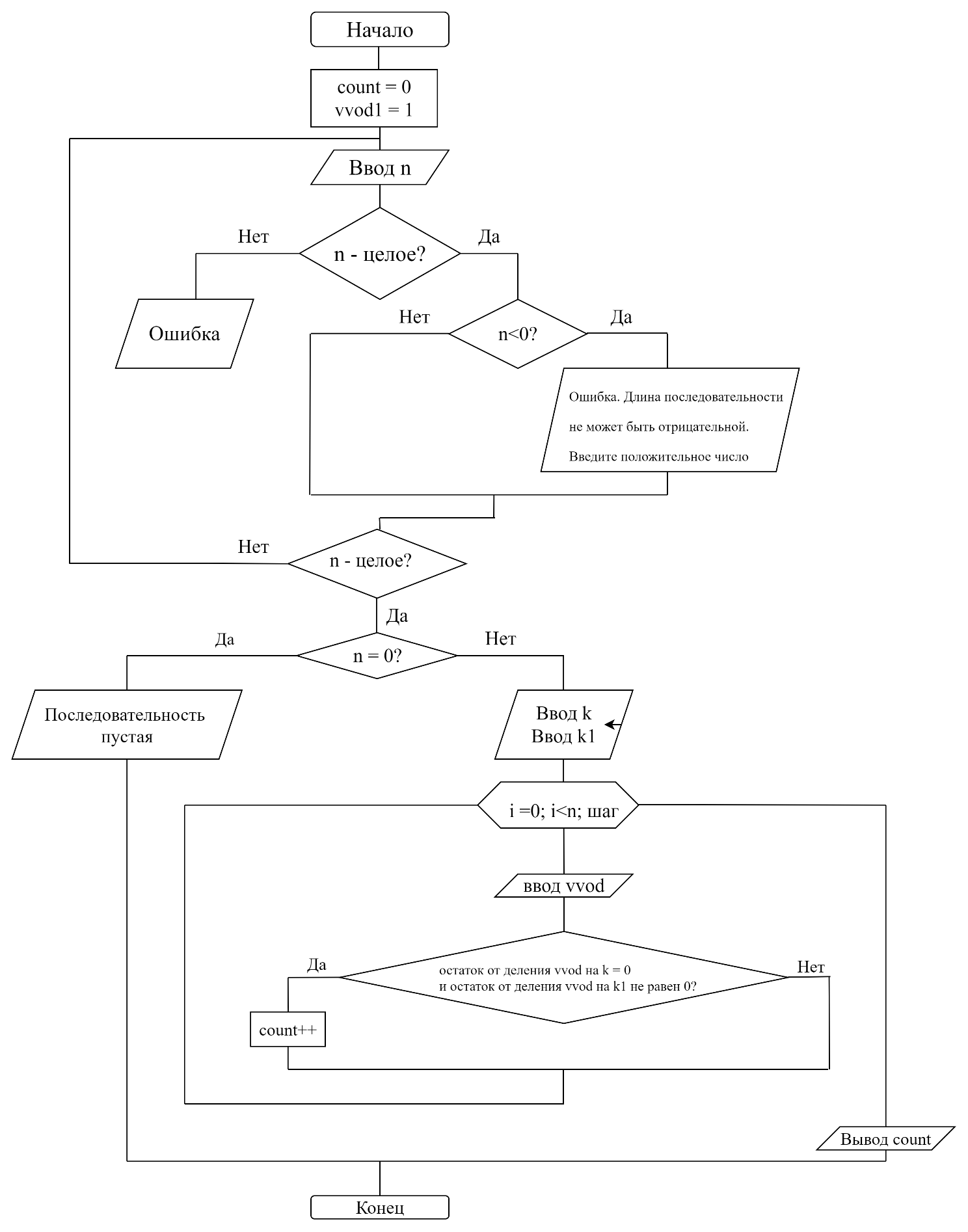
Анализ решения:

– Используется цикл с параметром, так как известно количество элементов последовательности.

Таблица 1 – Анализ условия для первой задачи

|  |  |  |
| --- | --- | --- |
| Исходные данные | Классы | Ожидаемый результат |
| n | n>0 | count (при правильном вводе остальных переменных) |
| n – не целое | Ошибка |
| n=0 | Последовательность пустая |
| n<0 | Ошибка |
| a | Целое число | count (при правильном вводе остальных переменных) |
| Не целое число | Ошибка |

|  |  |  |
| --- | --- | --- |
| k1 | Целое число | count (при правильном вводе остальных переменных) |
| Не целое число | Ошибка |
| k2 | Целое число | count (при правильном вводе остальных переменных) |
| Не целое число | Ошибка |

**Алгоритм**

**Рисунок 1 – Блок-схема к первой задаче**

**Программа**

using System;

namespace Лаба\_2

{

class Program

{

static void Main(string[] args)

{

int n; int k; int k1; int count = 0; int vvod; int vvod1 = 1;

Console.WriteLine("Введите количество чисел в последовательности");

string bug = Console.ReadLine();

while (!int.TryParse(bug, out n))

{

Console.WriteLine("Введите число!");

bug = Console.ReadLine();

}

n = Convert.ToInt32(bug);

if (n == 0)

Console.WriteLine("Последовательность пустая");

else if (n < 0)

{

do

{

Console.WriteLine("Ошибка. Длина последовательности не может быть отрицательна, введите " +

"положительное число");

bug = Console.ReadLine();

n = Convert.ToInt32(bug);

} while (n < 0);

}

if (n != 0) // n != 0

{

Console.WriteLine("Введите k");

bug = Console.ReadLine();

while (!int.TryParse(bug, out k))

{

Console.WriteLine("Введите целое число!");

bug = Console.ReadLine();

}

k = Convert.ToInt32(bug);

Console.WriteLine("Введите число k1");

bug = Console.ReadLine();

while (!int.TryParse(bug, out k1))

{

Console.WriteLine("Введите число!");

bug = Console.ReadLine();

}

k1 = Convert.ToInt32(bug);

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

{

Console.WriteLine("Введите {0} числo", vvod1++);

bug = Console.ReadLine();

while (!int.TryParse(bug, out vvod))

{

Console.WriteLine("Введите число!");

bug = Console.ReadLine();

}

vvod = Convert.ToInt32(bug);

if (vvod % k == 0 && vvod % k1 != 0)

{

count += 1;

}

}

Console.WriteLine("Количество элементов последовательности" +

" кратных числу {0} и не кратных числу {1} = {2}", k, k1, count);

}

}

}

}

**Тесты**

Таблица 2 – Тесты к первой задаче

|  |  |  |  |
| --- | --- | --- | --- |
| Тест, № | Входные данные | Ожидаемый результат | Выходные данные |
| 1 | n = 0 | Последовательность пустая | Последовательность пустая |
| 2 | n = -4 | Ошибка. Длина последовательности не может быть отрицательна, введите положительное число | Ошибка. Длина последовательности не может быть отрицательна, введите положительное число |
| 3 | n = 2  k = 4  k1 = 1  a1 (vvod) = 4  a2 (vvod) = 4 | Количество элементов последовательности кратных числу 4 и не кратных числу 1 = 0 | Количество элементов последовательности кратных числу 4 и не кратных числу 1 = 0 |
| 4 | n = 2  k = -1  k1 = 2  a1 (vvod) = 5  a2 (vvod) = =7 | Количество элементов последовательности кратных числу -1 и не кратных числу 2 = 2 | Количество элементов последовательности кратных числу -1 и не кратных числу 2 = 2 |
| 5 | n = 2  k = 3  k1 = 4  a1 (vvod) = 7  a2 (vvod) = 4 | Количество элементов последовательности кратных числу 3 и не кратных числу 4 = 0 | Количество элементов последовательности кратных числу 3 и не кратных числу 4 = 0 |
| 6 | n = 5,4 | Ошибка | Ошибка |
| 7 | n = 3  k = 4,4 | Ошибка | Ошибка |
| 8 | n = 2  k = 2  k1 = 3  a1 (vvod) = 4,1 | Ошибка | Ошибка |

Таблица 3 – Черный ящик к первой задаче

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Критерии | Т1 | Т2 | Т3 | Т4 | Т5 | Т6 | Т7 | Т8 |
| Последовательность пустая | + |  |  |  |  |  |  |  |
| Длина последовательности – отрицательное число |  | + |  |  |  |  |  |  |
| k1 = 1 |  |  | + |  |  |  |  |  |
| k или k1 отрицательно |  |  |  | + |  |  |  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Критерии | Т1 | Т2 | Т3 | Т4 | Т5 | Т6 | Т7 | Т8 |
| Количество элементов последовательности, кратных k и не кратных k1 = 0 |  |  | + |  | + |  |  |  |
| Количество элементов последовательности, кратных k и не кратных k1 > 0 |  |  |  | + |  |  |  |  |
| n – не целое |  |  |  |  |  | + |  |  |
| k – не целое |  |  |  |  |  |  | + |  |
| vvod – не целое |  |  |  |  |  |  |  | + |

Таблица 4 – МГТ к первой задаче

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Операторы | Т1 | Т2 | Т3 | Т4 | Т5 | Т6 | Т7 | Т8 |
| if (n == 0){} | + |  |  |  |  |  |  |  |
| else if (n < 0){}  do{} while (n < 0) |  | + |  |  |  |  |  |  |
| if (n != 0){} |  |  | + | + | + | + | + | + |
| for (int i = 0; i < n; шаг){} |  |  | + | + | + |  |  | + |
| if (vvod % k == 0 && vvod % k1 != 0) {} |  |  | + | + | + |  |  |  |

**Задача 2**

Дана последовательность целых чисел, за которой следует 0. Найти сумму четных элементов этой последовательности.

**Анализ**

Исходные данные:

– a1, a2, a3 – Элементы последовательности. (целые числа)

Выходные данные:

– summa – Сумма четных элементов последовательности. (целое число)

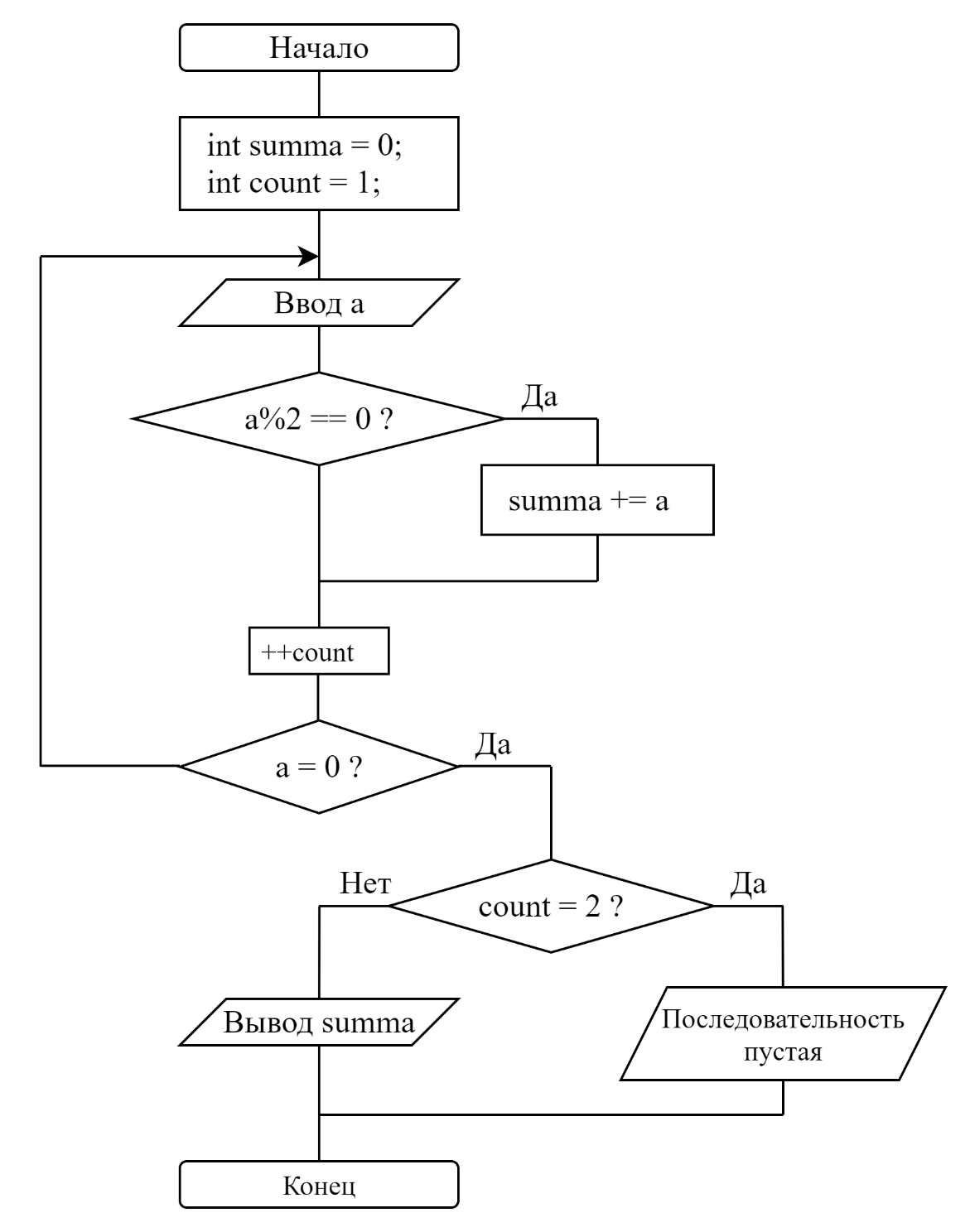
Анализ решения:

– используется цикл с условием, т. к. известно, что признаком окончания последовательности является 0.

Таблица 5 – Анализ условия для второй задачи

|  |  |  |
| --- | --- | --- |
| Исходные данные | Классы | Ожидаемый результат |
| a | Не целое число или не число | Ошибка |
| Целые числа, не равные 0 (если в конце следует ноль) | summa |
| Первое число – ноль | Пустая последовательность |

**Алгоритм**



**Рисунок 2 – Блок-схема ко второй задаче**

**Программа**

using System;

namespace Лаба\_2

{

class Program

{

static void Main(string[] args)

{

int summa = 0;

int count = 1;

int c;

int a;

do

{

Console.WriteLine("Введите {0} число последовательности", count);

string bug = Console.ReadLine();

while (!int.TryParse(bug, out a))

{

Console.WriteLine("Введите число!");

bug = Console.ReadLine();

}

a = Convert.ToInt32(bug);

if (a % 2 == 0)

summa += a;

++count;

c = a;

} while (c != 0);

if (count == 2)

Console.WriteLine("Последовательность пустая");

else

Console.WriteLine("Сумма четных элементов последовательности равна " + summa);

}

}

}

**Тесты**

Таблица 6 – Тесты ко второй задаче

|  |  |  |  |
| --- | --- | --- | --- |
| Тест, № | Входные данные | Ожидаемый результат | Выходные данные |
| 1 | a1 = 2  a2 = 3  a3 = 1  a4 = 4  a5 = 0 | Сумма четных элементов последовательности равна 6 | Сумма четных элементов последовательности равна 6 |
| 2 | a1 = 0 | Последовательность пустая | Последовательность пустая |
| 3 | a1 = 5,4 | Ошибка | Ошибка |

|  |  |  |  |
| --- | --- | --- | --- |
| Критерии | Т1 | Т2 | Т3 |
| Последовательность пустая |  | + |  |
| Элементы последовательности введены корректно | + |  |  |
| Элементы последовательности не целые числа |  |  | + |

Таблица 7 – Тестирование черного ящика

Таблица 8 – МГТ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Операторы | | Т1 | Т2 | Т3 |
| if (a % 2) == 0 {} | | + |  |  |
| do{} while (c != 0) | | + |  |  |
| if (count == 2) {} | Да |  | + |  |
| Нет | + |  |  |

**Задача 3**

Сформировать все числа Фибоначчи, не превышающие заданное число Q.

**Анализ**

Исходные данные:

1) Q – число – ограничитель. (тип – double).

Выходные данные:

1) a1, a2, a3…an – числа Фибоначчи, не превышающие заданное число Q (целые числа).

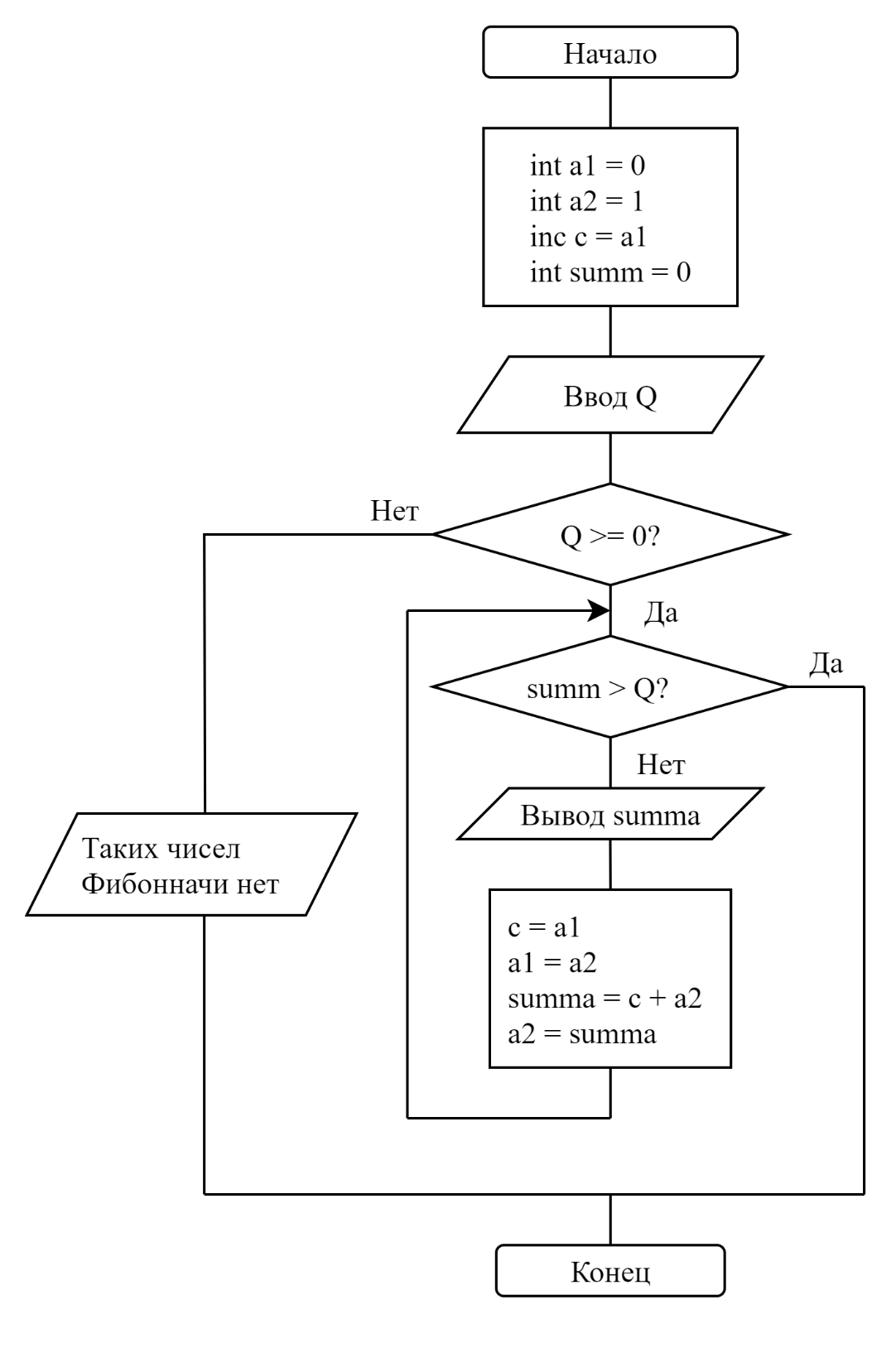
Анализ решения:

– используется цикл с предусловием, так как известно число-ограничитель.

Таблица 9 – Анализ условия для третьей задачи

|  |  |  |
| --- | --- | --- |
| Исходные данные | Классы | Ожидаемый результат |
| Q | Положительное число или ноль | Числа Фибоначчи, не превышающие заданное число |
| Отрицательное число | Таких чисел Фибоначчи нет |
| Не число | Ошибка ввода |

**Алгоритм**



**Рисунок 3 – Блок-схема к третьей задаче**

**Программа**

using System;

namespace Лаба\_2

{

class Program

{

static void Main(string[] args)

{

int a1 = 0;

int a2 = 1;

int c = a1;

int summ = 0;

double Q;

Console.WriteLine("Введите число Q. Числа Фибоначчи не будут превышать это число.");

string bug = Console.ReadLine();

while (!Double.TryParse(bug, out Q))

{

Console.WriteLine("Введите число!");

bug = Console.ReadLine();

}

Q = Convert.ToDouble(bug);

if (Q < 0)

Console.WriteLine("Таких чисел Фибоначчи нет");

else

Console.WriteLine("\nЧисла Фибоначчи представлены в следующем ряду");

{

while (summ <= Q)

{

Console.WriteLine(summ);

c = a1;

a1 = a2;

summ = c + a2;

a2 = summ;

}

}

}

}

}

**Тесты**

Таблица 10 – Тесты к третьей задаче

|  |  |  |  |
| --- | --- | --- | --- |
| Тест, № | Входные данные | Ожидаемый результат | Выходные данные |
| 1 | -1 | Таких чисел Фибоначчи нет | Таких чисел Фибоначчи нет |
| 2 | 4 | Числа Фибоначчи представлены в следующем ряду  0  1  2  3 | Числа Фибоначчи представлены в следующем ряду  0  1  2  3 |
| 3 | 0 | Числа Фибоначчи представлены в следующем ряду  0 | Числа Фибоначчи представлены в следующем ряду  0 |
| 4 | 5,6 | Числа Фибоначчи представлены в следующем ряду  0  1  2  3  5 | Числа Фибоначчи представлены в следующем ряду  0  1  2  3  5 |
| 5 | Абщихба | Ошибка | Ошибка |

Таблица 11 – Тестирование черного ящика

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Критерии | Т1 | Т2 | Т3 | Т4 | Т5 |
| Число Q отрицательно | + |  |  |  |  |
| Число Q положительно |  | + |  | + |  |
| Q = 0 |  |  | + |  |  |
| Q – не целое |  |  |  | + |  |
| Q – не число |  |  |  |  | + |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Операторы | | Т1 | Т2 | Т3 | Т4 | Т5 |
| if (Q < 0){} | да | + |  |  |  |  |
| нет |  | + | + | + |  |
| While (sum <= Q){} | |  | + | + | + |  |

Таблица 12 – МГТ