Пермский филиал федерального государственного автономного образовательного учреждения высшего образования

«Национальный исследовательский университет

«Высшая школа экономики» Пермь»

Факультет экономики, менеджмента и бизнес-информатики.

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1

студента образовательной программы «Программная инженерия»

по направлению подготовки *09.03.04 Программная инженерия*

Чепокова Елизара Сергеевича

Руководитель:

Кандидат технических наук

О.Л. Викентьева

Пермь, 2018 год

# **1. Постановка задачи:**

1. Постановка задач для конкретного варианта.
2. Анализ (классы входных и выходных данных, метод решения).
3. Алгоритм решения каждой задачи в виде блок-схемы.
4. Программы для решения задач на языке C#.
5. Тесты (с проверкой достаточности по критериям черного ящика и МГТ) для каждой задачи.

|  |  |  |  |
| --- | --- | --- | --- |
| № варианта | Задача 1 (6) | Задача 2 (28) | Задача 3 (57) |
| 23 | Дана последовательность из n целых чисел. Найти минимальный элемент в этой последовательности. | Дана последовательность целых чисел, за которой следует 0. Найти разность минимального и максимального элементов в этой последовательности. | , где n>2. |

# **2. Анализ задач:**

## **1 задача**

1. Задать длину последовательности
2. Ввод элементов последовательности
3. Вывод минимального элемента последовательности

|  |  |  |  |
| --- | --- | --- | --- |
| Входные данные | Классы входных данных | Выходные данные | Классы выходных данных |
| n - целое число | n<=0 | Сообщение об ошибке | Длина не может быть меньше или равна 0 |
| n - целое число,  а1 – целое число | n=1,  а – любое целое число | 1 | единица |
| n - целое число,  а1, …, an – целые числа,  min – первое число | n>1  а1, …, an – целые числа | Целое число от 1 до n | Min – минимальное число последовательности |

## **2 задача**

1. Ввод целых чисел последовательности
2. Ввод конца последовательности – 0
3. Сравнение минимального и максимального значения
4. Вывод разности минимального и максимального значения

|  |  |  |  |
| --- | --- | --- | --- |
| Входные данные | Классы входных данных | Выходные данные | Классы выходных данных |
| num1, …, num1, 0 | num1 – целые числа не равные 0 | r – целое число | r = max - min |
| num1 | num1 = 0 | Сообщение об ошибке | Последовательность пуста |

## **3 задача**

1. Задать число n, где n>2
2. Вывод ответа 

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Входные данные | Классы входных данных | Выходные данные | Классы выходных данных | Математическая модель |
| n ≤ 2 | a – целое положительное число  b – вещественное положительное число | Ответ в double= 0,999999694344575  Ответ в float= 0,006357849 | Вещественные положительные числа |  |

# **3. Алгоритм решения:**

## **Блок-схема для 1 задачи**

НЕТ

НЕТ

ДА

Начало

Ввод n (длина)

n - целое?

 Длина не может быть меньше или равна 0

n ≥ 0?

Ввод min

(1 число)

Ввод an

a < min?

 min = a

Вывод min

ДА

Конец

ДА

НЕТ

## **Блок-схема для 2 задачи**

НЕТ

ДА

ДА

ДА

ДА

НЕТ

ДА

Начало

Ввод num1

num1 - вещественное?

num1 = 0?

Вывод “Последовательность пуста”

Цикл, пока num1 != 0

Ввод num1

num1 - вещественное?

min = num1

max = num1

min > num1?

max < num1?

min = num1

НЕТ

min = num1

Конец цикла

Ввод num1 = 0

r = max-min

Вывод r

Конец

## **Блок-схема для 3 задачи**

НЕТ

НЕТ

ДА

Начало

Ввод n (длина)

n - целое?

n ≤ 2?

P =

Вывод P

ДА

Конец

# **4. Код программы:**

## **Код 1 программы:**

namespace ConsoleApplication1

{

class Program

{

static void Main(string[] args)

{

int n;

double min = 0, a;

bool elem;

Console.WriteLine("Введите длину последовательности");

do

{

elem = int.TryParse(Console.ReadLine(), out n);

if (n <= 0)

elem = false;

if (elem == false)

Console.WriteLine("Длина не может быть меньше или равна 0 \nВведите длину последовательности");

}

while (!elem);

elem = false;

Console.WriteLine("Введите элемент последовательности");

while (!elem)

{

elem = double.TryParse(Console.ReadLine(), out min);

if (elem == false)

Console.WriteLine("Элемент должен быть числом");

}

for (int i = 1; i < n; i++)

{

Console.WriteLine("Введите элемент последовательности");

do

{

elem = double.TryParse(Console.ReadLine(), out a);

if (elem == false)

Console.WriteLine("Элемент должен быть числом");

}

while (!elem);

if (a < min)

min = a;

}

Console.WriteLine(" ");

Console.WriteLine(" ");

Console.WriteLine($"Минимальный элемент последовательности равен {min}");

Console.ReadKey();

}

}

}

## **Код 2 программы:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApplication2

{

class Program

{

static void Main(string[] args)

{

int num1 = 0,

min = 0,

max = 0,

i = 0;

bool ok = false;

while (!ok)

{

try

{

Console.WriteLine("Введите 1 число");

num1 = Convert.ToInt32(Console.ReadLine());

ok = true;

}

catch (FormatException)

{

Console.WriteLine("Ошибка, повторите ввод");

ok = false;

}

}

min = num1;

max = num1;

while (num1 != 0)

{

try

{

i++;

if (min > num1)

min = num1;

if (max < num1)

max = num1;

Console.WriteLine("Введите {0} число", i + 1);

num1 = Convert.ToInt32(Console.ReadLine());

}

catch (FormatException)

{

Console.WriteLine("Ошибка, повторите ввод");

i--;

}

}

int r = max - min;

Console.WriteLine(" ");

Console.WriteLine(" ");

if (i == 0) Console.WriteLine("Последовательность пуста");

else Console.WriteLine("Разность минимума и максимума = " + r);

Console.ReadKey();

}

}

}

## **Код 3 программы:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApplication3

{

class Program

{

static void Main(string[] args)

{

int n;

double P = 1;

bool ok;

Console.WriteLine("Введите n:");

do

{

ok = int.TryParse(Console.ReadLine(), out n);

if (n <= 2)

ok = false;

if (ok == false)

Console.WriteLine("Введите n заново:");

}

while (!ok);

for (int i = 2; i <= n; i++)

{

i++;

P \*= 1 - (1 /(double)(i \* i));

}

Console.WriteLine(" ");

Console.WriteLine(" ");

Console.WriteLine("P = " + P);

Console.ReadKey();

}

}

}

# **5. Система тестов:**

## **Тесты для 1 задачи:**

### **Чёрный ящик**

Тесты

|  |  |  |  |
| --- | --- | --- | --- |
| № | Входные данные | Ожидаемый результат | Полученный результат |
| T1 | n=-5 | Ошибка | Ошибка |
| T2 | n=0 | Ошибка | Ошибка |
| T3 | n=1, a=5 | min = 5 | min = 5 |
| T4 | n=3, a: -2; 6; 9 | min = -2 | min = -2 |
| T5 | n=5, a: 1,9; 2; 3; 4; 5 | min = 1,9 | min = 1,9 |
| T6 | n=3, a: 99; 0; 1 | min = 0 | min = 0 |
| T7 | n=1, a=asd | Ошибка | Ошибка |

Критерии

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № | Проверяемая ситуация | Тесты | | | | | | |
| T1 | T2 | T3 | T4 | T5 | T6 | T7 |
| 1 | Входные данные для n | | | | | | | |
| 1.1 | n<0 | + |  |  |  |  |  |  |
| 1.2 | n=0 |  | + |  |  |  |  |  |
| 1.3 | n=1 |  |  | + |  |  |  | + |
| 1.4 | n>1 |  |  |  | + | + | + |  |
| 2 | Входные данные для a | | | | | | | |
| 2.1 | Ничего не введено | + | + |  |  |  |  |  |
| 2.2 | Ноль |  |  |  |  |  | + |  |
| 2.3 | Целые числа |  |  | + | + |  |  |  |
| 2.4 | Вещественные числа |  |  |  |  | + |  |  |
| 2.5 | Не числа |  |  |  |  |  |  | + |
| 3 | Выходные данные | | | | | | | |
| 3.1 | Ошибка | + | + |  |  |  |  | + |
| 3.2 | Целое число |  |  | + | + |  | + |  |
| 3.3 | Вещественное число |  |  |  |  | + |  |  |

### **Белый ящик**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | T1 | T2 | T3 | T4 | T5 | T6 | T7 |
| do {}while (!elem); //n | 0 |  | + |  |  |  |  |  |
| 1 |  |  | + |  |  |  | + |
| n | + |  |  | + | + | + |  |
| if (n <= 0) | + | + | + |  |  |  |  |  |
| - |  |  | + | + | + | + | + |
| while (!elem) {} //min | 0 |  |  |  |  |  | + |  |
| 1 |  |  |  |  |  | + |  |
| n |  |  | + | + | + | + |  |
| литер |  |  |  |  |  |  | + |
| if (elem == false) | + |  |  |  |  |  |  | + |
| - | + | + | + | + | + | + |  |
| for (int i = 1; i < n; i++) | + |  |  |  | + | + | + |  |
| - | + | + | + |  |  |  | + |
| do {}while (!elem); //a | 0 |  |  |  |  |  | + |  |
| 1 |  |  |  |  |  | + |  |
| n |  |  | + | + | + | + |  |
| if (a < min) | + |  |  |  |  |  | + |  |
| - | + | + | + | + | + |  | + |

## **Тесты для 2 задачи:**

### **Чёрный ящик**

Тесты

|  |  |  |  |
| --- | --- | --- | --- |
| № | Входные данные | Ожидаемый результат | Полученный результат |
| T1 | a: 0 | Последовательность пуста | Последовательность пуста |
| T2 | a: 1; 0 | r = 0 | r = 0 |
| T3 | a: 1; 2; 3; 4; 5; 6; 7; 8; 9; 0 | r = 8 | r = 8 |
| T4 | a: 1,2; 2,5; 3; 4,3; 5; 0 | Ошибка | Ошибка |
| T5 | a: 99; 2; 4; -99; 0 | r = 198 | r = 198 |
| T6 | a: 9; 2; 0 | r = 7 | r = 7 |
| T7 | a: asd; 3; 0 | Ошибка | Ошибка |

Критерии

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № | Проверяемая ситуация | Тесты | | | | | | |
| T1 | T2 | T3 | T4 | T5 | T6 | T7 |
| 1 | Входные данные для a | | | | | | | |
| 1.1 | Ноль | + |  |  |  |  |  |  |
| 1.2 | Единица |  | + | + |  |  |  |  |
| 1.3 | Целые числа |  |  |  | + |  | + |  |
| 1.4 | Вещественные числа |  |  |  |  | + |  |  |
| 1.5 | Не числа |  |  |  |  |  |  | + |
| 3 | Выходные данные | | | | | | | |
| 3.1 | Целые числа |  | + | + |  | + | + |  |
| 3.2 | Ошибка | + |  |  | + |  |  | + |

### **Белый ящик**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | T1 | T2 | T3 | T4 | T5 | T6 | T7 |
| while (!ok) {} //num1 | 0 | + |  |  |  |  |  |  |
| 1 |  | + | + | + |  |  |  |
| n |  |  |  |  | + | + |  |
| литер |  |  |  |  |  |  | + |
| try {} catch (FormatException) {} | 0 | + |  |  |  |  |  |  |
| 1 |  | + | + | + |  |  |  |
| n |  |  |  |  | + | + |  |
| литер |  |  |  |  |  |  | + |
| while (num1 != 0) //num1 | 0 | + | + | + | + | + | + | + |
| 1 |  | + | + | + |  |  |  |
| n |  |  |  |  | + | + | + |
| try {} catch (FormatException) {} | 0 | + | + | + | + | + | + | + |
| 1 |  | + | + | + |  |  |  |
| n |  |  |  |  | + | + | + |
| if (min > num1) | + |  |  |  |  | + | + |  |
| - | + | + | + | + |  |  |  |
| if (max < num1) | + |  | + | + | + | + |  |  |
| - | + |  |  |  |  | + | + |
| if (i == 0) | + | + |  |  |  |  |  |  |
| - |  | + | + | + | + | + | + |

## Тесты для 3 задачи:

Тесты

|  |  |  |  |
| --- | --- | --- | --- |
| № | Входные данные | Ожидаемый результат | Полученный результат |
| T1 | n: 0 | Ошибка | Ошибка |
| T2 | n: 1 | Ошибка | Ошибка |
| T3 | n: 4 | P = 0,853333333333333 | P = 0,853333333333333 |

Критерии

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Проверяемая ситуация | Тесты | | |
| T1 | T2 | T3 |
| 1 | Входные данные для n | | | |
| 1.1 | Ноль | + |  |  |
| 1.2 | Единица |  | + |  |
| 1.3 | Целые числа |  |  | + |
| 3 | Выходные данные | | | |
| 3.1 | Вещественные числа |  |  | + |
| 3.2 | Ошибка | + | + |  |

### **Белый ящик**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | T1 | T2 | T3 |
| while (!ok) {} //num1 | 0 | + |  |  |
| 1 |  |  |  |
| n |  |  |  |
| литер |  |  |  |
| if (n <= 2) | + |  |  |  |
|  | - |  |  |  |
| if (ok == false) | + |  |  |  |
|  | - |  |  |  |
| for (int i = 2; i <= n; i++) | + |  |  |  |
|  | - |  |  |  |