**Пермский филиал федерального государственного автономного образовательного учреждения высшего**

**“Национальный исследовательский университет**

**“Высшая школа экономики”**

Факультет экономики, менеджмента и бизнес-информатики

Дмитриев Арсений Алексеевич

# Лабораторная работа №6. «Класс Array. Строки. Класс String»

Отчет по практической работе

**студента образовательной программы «Программная инженерия»**

**по направлению подготовки *09.03.04 Программная инженерия***

**руководитель**

**к. тех. н., доцент каф.**

**инф. техн. в биз.**

**Викентьева О.Л.**

**Пермь, 2019 г.**

**Постановка задачи**

**Общая:**

Постановка задачи 1.

1. Создать динамический массив (одномерный, двумерный, рваный) из элементов заданного типа. При заполнении массива использовать 2 способа (ручной и с помощью ДСЧ).
2. Массив вывести на печать.
3. Выполнить операции с массивом, указанные в варианте, используя, по возможности, методы класса Array.
4. Результаты обработки вывести на печать.

Постановка задачи 2.

1. Ввести строку символов с клавиатуры. Строка состоит из слов, разделенных пробелами (пробелов может быть несколько) и знаками препинания (, ;:). В строке может быть несколько предложений, в конце каждого предложения стоит знак препинания (.!?).
2. Выполнить обработку строки в соответствии с вариантом.
3. Результаты обработки вывести на печать.

**Вариант №7**

**Задача 1:** В одномерном массиве из вещественных (double) чисел удалить все элементы кратные минимальному элементу массива.

**Задача 2:** Сдвинуть циклически влево каждое слово на количество символов равное номеру этого слова в строке.

**Анализ**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Функция | Входные данные | Классы входных данных | Выходные данные |
| Основные функции | | | | |
| 1 | Формирование одномерного массива | Размер массива | Целое число >=1 | Одномерный массив |
| Целое число <=0 | Сообщение об ошибке |
| 1.1 | Формирование одномерного массива с помощью ДСЧ | Вещественное число |
| Нечисло |
| 1.2 | Формирование одномерного массива с клавиатуры | Элементы массива | Целое число | Одномерный массив |
| Вещественное число |
| Нечисло | Сообщение об ошибке |
| 2 | Удаление из одномерного массива элементов кратных минимальному | * Массив * Размер массива | Непустой массив, Элементов кратных минимальному >1, но <размера массива | Измененный массив |
|
|
| Непустой массив, Элементов кратных минимальному = 1 (т.е. только минимальный эл.) |
|
|
| Непустой массив, все элементы кратны минимальному | Пустой массив |
|
|
| Пустой массив | Сообщение об ошибке |
| 3 | Ввод строки с клавиатуры | Строка | Непустая строка | Строка |
| Пустая строка | Сообщение об ошибке |
| 4 | Формирование строки из массива слов | Количество слов | Целое число >=1 | Строка |
| Целое число <=0 | Сообщение об ошибке |
| Вещественное число |
| Нечисло |
| 5 | Циклический сдвиг влево на K символов | Строка | Непустая строка | Измененная строка |
| Пустая строка | Сообщение об ошибке |
| Число сдвигов | Целое неотрицательное число | Измененная строка |
| Целое отрицательное | Сообщение об ошибке |
| Вещественное число |
| Нечисло |
| Вспомогательные функции | | | | |
| 6 | Ввод целого числа | * Строка, введенная пользователем * Нижняя граница допустимого диапазона * Верхняя граница диапазона | Введенная строка - целое число попадающее в допустимый диапазон от нижней до верхней границы | Целое число |
|
|
|
| Строка - не целое число | Сообщение об ошибке |
| Строка - нечисло |
| Строка - целое число, не попадающее в допустимый диапазон |
|
|
| Строка - целое число, > 2^31-1 или <-2^31 |
|
| 7 | Ввод вещественного числа | * Строкавведенная пользователем * Нижняя граница допустимого диапазона * Верхняя граница диапазона | Введенная строка - целое/вещественное число попадающее в допустимый диапазон от нижней до верхней границы | Вещественное число |
|
|
|
| Строка - не целое/вещественное число | Сообщение об ошибке |
| Строка - нечисло |
| Строка - вещественное число, не попадающее в допустимый диапазон |
|
|
| Строка - вещественное число, не попадающее в поддерживаемый диапазон double |
|
| 8 | Поиск минимального элемента массива | Массив | Непустой массив | Минимальный элемент |
| Пустой массив | Сообщение об ошибке |
| 9 | Печать диалогового меню для пользователя |  |  |  |
|

**Проектирование**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Функция | Входные данные | Выходные данные |
| Основные функции | | | |
| 1 | ReadArray()  Формирование массива с клавиатуры | int sizeOfArray - размер массива | double[] arr - массив |
|
| 2 | CreateRandomArray() Формирование массива, с помощью ДСЧ | int sizeOfArray - размер массива | double[] arr - массив |
|
| 3 | ArrayDeleteMinMultiply()  Удаление элементов, кратных минимальному | double[] arr - массив | double[] newArr - измененный массив |
|
| 4 | GetText()  Ввод строки с клавиатуры |  | string buf - введенная непустая строка |
|
| 5 | CreateNewText()  Формирование строки из имеющихся слов | int size - количество слов в строке | string txt - сформированная строка |
|
| 6 | LeftShift()  Циклический сдвиг влево | string word - сдвигаемая строка, int k - количество сдвигов | string txt - измененная строка |
|
| Вспомогательные функции | | | |
| 7 | ReadInteger()  Ввод целого числа | int lowerBound - нижняя граница диапазона, int upperBound - верхняя граница диапазона | int number - целое число |
|
| 8 | ReadDouble()  Ввод вещественного числа | int lowerBound - нижняя граница диапазона, int upperBound - верхняя граница диапазона | double number - вещественное число |
|
| 9 | ArrayFindMin()  Поиск минимального элемента | double[] arr - массив | double min - значение минимального элемента |
|
| 10 | MainMenu(), Task1Menu(), Task2Menu()  Печать диалогового меню для пользователя |  |  |
|

**Текст программы**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Laba6

{

class Program

{

const double MIN\_ELEM = 0.0;

const double MAX\_ELEM = 1000.0;

const int MIN\_SIZE = 1;

const int MAX\_SIZE = 1000;

static string[] RND\_WORDS = { "Я", "памятник", "себе", "воздвиг", "нерукотворный", "Вознесся", "выше", "он", "главою", "непокорной",

"Александрийского", "столпа", "Жив", "будет", "хоть", "один", "пиит", "Веленью", "божию", "муза", "будь", "послушна", "Обиды" };

static char[] PUNCT = { '.', ',', '!', '?', ':', ';' };

static string DIVIDER = "\n------------------------------------------------------------\n";

#region Task1

static double ReadDouble(string userInstruction, double lowerBound = MIN\_ELEM, double upperBound = MAX\_ELEM)

{

double number = 0;

bool ok = false;

do

{

try

{

Console.Write(userInstruction);

string input = Console.ReadLine();

number = Convert.ToDouble(input);

if (number >= lowerBound && number <= upperBound) ok = true;

else

{

Console.WriteLine("Ошибка: Введенное число должно попадать в промежуток от {0} до {1}", lowerBound, upperBound);

}

}

catch (FormatException)

{

Console.WriteLine("Ошибка: Введите вещественное число!");

}

catch (OverflowException)

{

Console.WriteLine("Ошибка: Вещественное число такого порядка не поддерживается!");

}

} while (!ok);

return number;

}

static double[] CreateRandomArray(int sizeOfArray)

{

double[] arr = new double[sizeOfArray];

Random random = new Random();

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

{

string txt = String.Format("{0:f2}", random.NextDouble() \* MAX\_ELEM);

arr[i] = double.Parse(txt);

}

return arr;

}

static double[] ReadArray(int sizeOfArray)

{

double[] arr = new double[sizeOfArray];

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

arr[i] = ReadDouble($"Введите {i+1} элемент: ", 0.0, 1000.0);

return arr;

}

static double ArrayFindMin(double[] arr)

{

double min = arr[0];

for (int i = 1; i < arr.Length; i++) if (arr[i] < min) min = arr[i];

return min;

}

static int ArraySizeOfNonMinMultiplyElem(double[] arr, double min)

{

int k = 0;

for (int i = 0; i < arr.Length; i++) if (arr[i] % min != 0) k++;

return k;

}

static double[] ArrayDeleteMinMultiply(double[] arr)

{

double min = ArrayFindMin(arr);

int newSize = ArraySizeOfNonMinMultiplyElem(arr, min);

double[] newArr = new double[newSize];

int sizeOfArray = arr.Length;

int index = 0;

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

{

if (arr[i] % min != 0) newArr[index++] = arr[i];

}

return newArr;

}

#endregion

#region Task2

static string LeftShift(string str, int k)

{

char[] txt = str.ToCharArray();

for(int times = 0; times < k; times++)

{

char tmp = txt[0];

for(int i = 0; i<txt.Length-1; i++)

{

txt[i] = txt[i + 1];

}

txt[txt.Length - 1] = tmp;

}

return (new string(txt));

}

static string[] ChangeWords(string[] arr)

{

for (int i = 0; i < arr.Length; i++) arr[i] = LeftShift(arr[i], i+1);

return arr;

}

static string GetChangedText(string txt)

{

string[] words = GetWordsFromText(txt);

words = ChangeWords(words);

string newTxt = "";

int len = 0;

int i = 0;

while(i < words.Length)

{

newTxt += words[i];

len += words[i].Length;

if (i == words.Length-1) {

while (len < txt.Length)

{

if(txt[len] != ' ') newTxt += txt[len];

len++;

}

}

else

{

// Нужно проверить не пришли ли мы к началу след.слова

// Для этого нужно найти индекс изначальной первой буквы слова в изменном слове

// Его можно найти по формуле: (Len - (i+1) % Len) % Len

// Где i - индекс след. слова, Len - его длина

while (txt[len] != words[i+1][(words[i+1].Length - (i + 2) % words[i+1].Length) % words[i+1].Length])

{

if (txt[len] != ' ') newTxt += txt[len];

len++;

}

newTxt += ' ';

}

i++;

}

return newTxt;

}

static string[] GetWordsFromText(string txt)

{

char[] sep = { ' ', '.', '!', '?', '-', ',', ':', ';'};

string[] words = txt.Split(sep, StringSplitOptions.RemoveEmptyEntries);

return words;

}

static string GetText(string userInstruction)

{

string buf;

do

{

Console.Write(userInstruction);

buf = Console.ReadLine();

if (buf.Length == 0) Console.WriteLine("Введите непустую строку!");

} while (buf.Length == 0);

return buf;

}

static string CreateNewText(int size)

{

int wLen = RND\_WORDS.Length;

int pLen = PUNCT.Length;

int[] used = new int[wLen];

Random rand = new Random();

string txt = "";

for(int i = 0; i<size; i++) {

int pos;

bool ok = false;

do

{

pos = rand.Next(0, wLen);

if (used[pos] < 2) ok = true;

} while (!ok);

used[pos]++;

txt += RND\_WORDS[pos];

int prob = rand.Next(0, 21);

if (prob >= 10 && prob <= 12)

{

pos = rand.Next(0, pLen);

txt += PUNCT[pos];

}

txt += ' ';

}

return txt;

}

#endregion

#region interface

static int ReadInteger(string userInstruction, int lowerBound = MIN\_SIZE, int upperBound = MAX\_SIZE)

{

int number = 0;

bool ok = false;

do

{

try

{

Console.Write(userInstruction);

string buf = Console.ReadLine();

number = Convert.ToInt32(buf);

if (number >= lowerBound && number <= upperBound) ok = true;

else Console.WriteLine("Ошибка: введенное число должно попадать в промежуток от {0} до {1}", lowerBound, upperBound);

}

catch (FormatException)

{

Console.WriteLine("Ошибка: введите целое число!");

}

catch (OverflowException)

{

Console.WriteLine("Ошибка: введенное число не поддерживается!");

}

} while (!ok);

return number;

}

static bool EmptyArraySituation(ref double[] arr, ref bool finishStage)

{

Console.WriteLine("После удаление всех кратных элементов массив стал пустым, выберите что делать дальше:");

Console.WriteLine(" 1: Сформировать новый массив с помощью ДСЧ");

Console.WriteLine(" 2: Сформировать новый массив с клавиатуры");

Console.WriteLine(" 0: Продолжить работу без нового массива и вернуться в основное меню");

int userChoise = ReadInteger("Введите выбранную опцию: ", 0, 2);

Console.WriteLine(DIVIDER);

switch (userChoise)

{

case 1:

{

int size = ReadInteger("Введите размер массива: ");

arr = CreateRandomArray(size);

return true;

}

case 2:

{

int size = ReadInteger("Введите размер массива: ");

arr = ReadArray(size);

return true;

}

default:

{

finishStage = true;

return false;

}

}

}

static void Task1Menu(bool arr)

{

Console.WriteLine("Доступные опции:");

Console.WriteLine($" 1: Сформировать массив с помощью ДСЧ");

Console.WriteLine($" 2: Сформировать массив с клавиатуры");

if(arr) Console.WriteLine($" 3: Удалить элементы, кратные минимальному");

Console.WriteLine($" 0: Вернуться в прошлое меню");

}

static void Task2Menu(bool str)

{

Console.WriteLine("Доступные опции:");

Console.WriteLine($" 1: Сформировать текст из стандратного набора слов");

Console.WriteLine($" 2: Ввести текст с клавиатуры");

if(str) Console.WriteLine($" 3: Сдвинуть циклически каждое слово в тексте влево");

Console.WriteLine($" 0: Вернуться в прошлое меню");

}

static void MainMenu()

{

Console.WriteLine("Доступные опции:");

Console.WriteLine($" 1: Работать с одномерным массивом");

Console.WriteLine($" 2: Работать со строкой");

Console.WriteLine($" 0: Завершить выполнение программы");

}

static void ArrayPrint(double[] arr)

{

Console.WriteLine("\nТекущий массив:");

for(int i = 0; i<arr.Length; i++)

{

Console.Write(arr[i] + " ");

}

}

static void ArrayHandler(ref double[] arr, ref bool arrFlag, ref bool finish)

{

int userChoise;

bool finishStage = false;

do

{

Console.WriteLine(DIVIDER);

Task1Menu(arrFlag);

userChoise = ReadInteger("Введите выбранную опцию: ", 0, arrFlag ? 3 : 2);

switch (userChoise)

{

case 0:

{

finishStage = true;

break;

}

case 1:

{

Console.WriteLine(DIVIDER);

int size = ReadInteger("Введите размер массива: ");

arr = CreateRandomArray(size);

arrFlag = true;

ArrayPrint(arr);

break;

}

case 2:

{

Console.WriteLine(DIVIDER);

int size = ReadInteger("Введите размер массива: ");

arr = ReadArray(size);

arrFlag = true;

ArrayPrint(arr);

break;

}

case 3:

{

Console.WriteLine(DIVIDER);

arr = ArrayDeleteMinMultiply(arr);

if (arr.Length == 0)

{

arrFlag = EmptyArraySituation(ref arr, ref finishStage);

}

ArrayPrint(arr);

break;

}

}

} while (!finishStage);

}

static void StringHandler(ref string txt, ref bool txtFlag, ref bool finish)

{

bool finishStage = false;

do

{

Console.WriteLine(DIVIDER);

Task2Menu(txtFlag);

int userChoise = ReadInteger("Введите выбранную опцию: ", 0, txtFlag ? 3 : 2);

switch (userChoise)

{

case 0:

{

finishStage = true;

break;

}

case 1:

{

Console.WriteLine(DIVIDER);

int size = ReadInteger("Введите количество слов, которые нужно использовать: ", 1, RND\_WORDS.Length \* 2 - 1);

txt = CreateNewText(size);

txtFlag = true;

Console.WriteLine($"Текущая строка:\n{txt}");

break;

}

case 2:

{

Console.WriteLine(DIVIDER);

txt = GetText("Введите текст: ");

txtFlag = true;

Console.WriteLine($"Текущая строка:\n{txt}");

break;

}

case 3:

{

Console.WriteLine(DIVIDER);

txt = GetChangedText(txt);

Console.WriteLine($"Текущая строка:\n{txt}");

break;

}

}

} while (!finishStage);

}

#endregion

static void Main(string[] args)

{

double[] arr = null;

string txt = null;

bool arrFlag = false, txtFlag = false;

int userChoise;

bool finish = false;

do

{

MainMenu();

userChoise = ReadInteger("Введите выбранную опцию: ", 0, 2);

if (userChoise == 1) ArrayHandler(ref arr, ref arrFlag, ref finish);

else if (userChoise == 2) StringHandler(ref txt, ref txtFlag, ref finish);

else finish = true;

} while (!finish);

}

}

}

**Тестирование**

1. Ввод целого числа

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Критерии\тесты | Т1 | Т2 | Т3 | Т4 |
| Критерии входных данных: введенная строка | | | | |
| это целое число, попадающее в диапазон | + | + | + | + |
| это целое число, не попадающее в диапазон |  |  | + |  |
| это вещественное число |  | + |  |  |
| это нечисло | + |  |  |  |
| Критерии выходных данных | | | | |
| Целое число | + | + | + | + |
| Сообщение об ошибке | + | + | + |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Диапазон | Ввод | Ожидаемый результат | Результат работы |
| Т1 | 1 - 1000 | ^t 5 | Сообщение об ошибке  Целое число (5) | Сообщение об ошибке  Целое число (5) |
| Т2 | 1 - 1000 | 1,586 1000 | Сообщение об ошибке  Целое число (1000) | Сообщение об ошибке  Целое число (1000) |
| Т3 | 1 - 1000 | -1 1001 1 | Сообщение об ошибке  Целое число (1) | Сообщение об ошибке  Целое число (1) |
| Т4 | 1 - 1000 | 25 | Целое число (25) | Целое число (25) |

1. Ввод вещественного числа

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Критерии\тесты | Т1 | Т2 | Т3 | Т4 |
| Критерии входных данных: введенная строка | | | | |
| это целое/вещественное число, попадающее в диапазон | + | + | + | + |
| это целое/вещественное число, не попадающее в диапазон |  | + | + |  |
| это нечисло | + |  | + |  |
| Критерии выходных данных | | | | |
| Вещественное число | + | + | + | + |
| Сообщение об ошибке | + | + | + |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Диапазон | Ввод | Ожидаемый результат | Результат работы |
| Т1 | 0,0 – 1000,0 | ^t 5,262 | Сообщение об ошибке  Вещественное число (5,262) | Сообщение об ошибке  Вещественное число (5,262) |
| Т2 | 0,0 – 1000,0 | -0,5 1,586 | Сообщение об ошибке  Вещественное число (1,586) | Сообщение об ошибке  Вещественное число (1,586) |
| Т3 | 0,0 – 1000,0 | 1103,32 $1 1 | Сообщение об ошибке  Вещественное число (1) | Сообщение об ошибке  Вещественное число (1) |
| Т4 | 0,0 – 1000,0 | 404,505 | Вещественное число (404,505) | Вещественное число (404,505) |

1. Формирование массива с помощью ДСЧ

|  |  |
| --- | --- |
| Критерии\тесты | Т1 |
| Критерии входных данных: размер массива | |
| Целое число >= 1 | + |
| Критерии выходных данных | |
| Массив из случайных чисел | + |

|  |  |  |  |
| --- | --- | --- | --- |
| № | Ввод | Ожидаемый результат | Результат работы |
| Т1 | 4 | Массив из 4 элементов | Массив из 4 элементов |

1. Формирование массива с клавиатуры

|  |  |
| --- | --- |
| Критерии\тесты | Т1 |
| Критерии входных данных: размер массива | |
| Целое число >= 1 | + |
| Критерии входных данных: элементы массива | |
| Вещественные числа из диапазона [0,0 ; 1000,0] | + |
| Критерии выходных данных | |
| Массив из случайных чисел | + |

|  |  |  |  |
| --- | --- | --- | --- |
| № | Ввод | Ожидаемый результат | Результат работы |
| Т1 | 6 2,3 12,1 11,7 3,9 39 0,17 | Массив {2,3;12,1;11,7;3,9;39;0,17} | Массив {2,3;12,1;11,7;3,9;39;0,17} |

1. Удаление элементов, кратных минимальному

|  |  |  |  |
| --- | --- | --- | --- |
| Критерии\тесты | Т1 | Т2 | Т3 |
| Критерии входных данных: массив | | | |
| Не пустой | + | + | + |
| Критерии длины набора: элементы кратные минимальному | | | |
| = 1 (сам минимальный элемент) |  | + |  |
| > 1 и < размера | + |  |  |
| = размер массива |  |  | + |
| Критерии выходных данных | | | |
| Измененный массив | + | + |  |
| "Массив стал пустым" |  |  | + |

|  |  |  |  |
| --- | --- | --- | --- |
| № | Ввод | Ожидаемый результат | Результат работы |
| Т1 | Массив {2; 3; 11,7; 8} | Массив {3; 11,7} | Массив {3; 11,7} |
| Т2 | Массив {1,01; 12,4; 7;14,1} | Массив {12,4; 7;14,1} | Массив {12,4; 7;14,1} |
| Т3 | Массив {1; 2,4; 4,8} | Пустой массив | Пустой массив |

1. Ввод строки с клавиатуры

|  |  |  |
| --- | --- | --- |
| Критерии\тесты | Т1 | Т2 |
| Критерии входных данных: строка | | |
| Пустая | + |  |
| Непустая | + | + |
| Критерии выходных данных | | |
| Строка | + | + |
| Сообщение об ошибке | + |  |

|  |  |  |  |
| --- | --- | --- | --- |
| № | Ввод | Ожидаемый результат | Результат работы |
| Т1 | « »  «Новая строка» | Сообщение об ошибке  «Новая строка» | Сообщение об ошибке  «Новая строка» |
| Т2 | «Мама мыла раму» | «Мама мыла раму» | «Мама мыла раму» |

1. Формирование строки из массива слов

|  |  |  |
| --- | --- | --- |
| Критерии\тесты | Т1 | Т2 |
| Критерии входных данных: количество слов | | |
| Кол-во слов >=1 | + |  |
| Критерии выходных данных | | |
| Строка | + | + |

|  |  |  |  |
| --- | --- | --- | --- |
| № | Ввод | Ожидаемый результат | Результат работы |
| Т1 | 10 | Строка из 10 слов | Строка из 10 слов |

1. Циклический сдвиг слов в строке

|  |  |
| --- | --- |
| Критерии\тесты | Т1 |
| Критерии входных данных: слово |  |
| Непустое | + |
| Критерии входных данных: число сдвигов |  |
| = длина слова или кратно длины слова | + |
| >=1 и < длины слова | + |
| >длины слова | + |
| Критерии выходных данных |  |
| Измененные слова | + |
| Слова без изменений | + |

|  |  |  |  |
| --- | --- | --- | --- |
| № | Ввод | Ожидаемый результат | Результат работы |
| Т1 | Робот упал днем в озеро Чад | оботР алуп мдне в озеро Чад | оботР алуп мдне в озеро Чад |