**Федеральное государственное автономное образовательное учреждение высшего образования   
"Национальный исследовательский университет   
"Высшая школа экономики"**

*Факультет социально-экономических и компьютерных наук*

Панфилов Даниил Сергеевич

**ЛАБОРАТОРНАЯ РАБОТА №6**

*Строки*

студента образовательной программы «Разработка информационных систем для бизнеса» по направлению подготовки *38.03.05 Бизнес-информатика*

Руководитель

ученая степень, ученое

звание, должность

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

И.О. Найданов

Пермь, 2023 год

# 1. Постановка задачи.

## 1.1. Общая постановка задачи.

1. Ввести строку символов (с клавиатуры или из массива заранее сформированных тестовых строк). Строка состоит из слов, разделенных пробелами (пробелов может быть несколько) и знаками препинания (, ;:). В строке может быть несколько предложений, в конце каждого предложения стоит один знак препинания (.!?).
2. Выполнить обработку строки в соответствии с вариантом, используя по возможности, методы класса String.
3. Результаты обработки вывести на печать.

## 1.2. Частная постановка задачи.

|  |  |  |
| --- | --- | --- |
| 2 | Перевернуть каждое четное слово. | Исходная строка: В лесу родилась елочка. В лесу она росла. Зимой и летом стройная, зеленая была.  Результат: В усел родилась акчоле. В усел она алсор. Зимой и летом яанйортс, зеленая алыб. |

# 2. Анализ задачи.

| **№** | **Описание функции** | **Входные данные** | **Классы входных данных** | **Классы выходных данных** |
| --- | --- | --- | --- | --- |
| **Основные функции** | | | | |
| 1 | Ввод строки | Текст | Строка | * Строка * Сообщение о том, что текст введён |
| 2 | Генерация случайной строки из имеющихся | Пустая строка | Пустая строка | * Строка * Сообщение о том, что выбран случайный текст из имеющегося списка |
| 3 | Переворот каждого чётного слова | Строка | Строка | * Строка с перевёрнутыми чётными словами * Сообщение о том, что текста нет * Сообщение о том, что чётные слова перевёрнуты |
| 4 | Печать строки | Строка | Строка | Вывод строки в консоль |

# 3. Проектирование.

## 3.1. Проектирование функций.

| **№** | **Описание функции** | **Входные данные** | **Классы входных данных** | **Классы выходных данных** |
| --- | --- | --- | --- | --- |
| **Основные функции** | | | | |
| 1 | InputStr()  Ввод строки | str - текст | Строка | * Строка * Сообщение о том, что текст введён |
| 2 | InputStrRnd()  Генерация случайной строки из имеющихся | str - пустая строка | Пустая строка | * Строка * Сообщение о том, что выбран случайный текст из имеющегося списка |
| 3 | ReverseEvenWord()  Переворот каждого чётного слова | str - строка | Строка | * Строка с перевёрнутыми чётными словами * Сообщение о том, что текста нет * Сообщение о том, что чётные слова перевёрнуты |
| 4 | PrintStr()  Печать строки | str - cтрока | Строка | Вывод строки в консоль |

## 3.2. Листинг программы.

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Reflection;

using System.Reflection.Emit;

using System.Text;

using System.Threading.Tasks;

namespace Лаба\_6

{

internal class Program

{

static double InputDoubleElement() // проверка на число (тип double)

{

bool isCorrert;

double number;

do

{

isCorrert = double.TryParse(Console.ReadLine(), out number);

if (!isCorrert) Console.WriteLine("Пожалуйста, введите целое число");

} while (!isCorrert);

return number;

}

static int InputIntNumber() // проверка на целое число

{

bool isCorrert;

int number;

do

{

isCorrert = int.TryParse(Console.ReadLine(), out number);

if (!isCorrert) Console.WriteLine("Пожалуйста, введите целое число");

} while (!isCorrert);

return number;

}

static void CreateMatr(ref double[,] matr) //создание двумерного массива вручную

{

int str = 0;

int col = 0;

do

{

Console.WriteLine("Введите количество строк");

str = InputIntNumber();

} while (str < 0);

do

{

Console.WriteLine("Введите количество столбцов");

col = InputIntNumber();

} while (col < 0);

matr = new double[str, col];

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

{

for (int j = 0; j < col; j++)

{

Console.WriteLine($"Введите элемент массива под номером: ({i + 1}; {j + 1})");

matr[i, j] = InputDoubleElement();

}

}

Console.WriteLine("Массив сформирован");

}

static void CreateMatrRnd(ref double[,] matr) //создание двумерного массива с помощью ДСЧ

{

int str = 0;

int col = 0;

do

{

Console.WriteLine("Введите количество строк");

str = InputIntNumber();

} while (str < 0);

do

{

Console.WriteLine("Введите количество столбцов");

col = InputIntNumber();

} while (col < 0);

matr = new double[str, col];

Random rnd = new Random();

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

{

for (int j = 0; j < col; j++)

{

matr[i, j] = rnd.Next(-10, 10);

}

}

Console.WriteLine("Массив сформирован");

}

static void PrintMatr(double[,] matr) // печать двумерного массива

{

if (IsEmptyMatr(matr)) Console.WriteLine("Массив пуст");

else

{

Console.WriteLine("Ваш массив:");

for (int i = 0; i < matr.GetLength(0); i++)

{

for (int j = 0; j < matr.GetLength(1); j++)

{

Console.Write(matr[i, j] + " ");

}

Console.WriteLine();

}

}

}

static bool IsEmptyMatr(double[,] matr) // проверка длины двумерного массива

{

if (matr == null || matr.Length == 0) return true;

else return false;

}

static void Commands1() // пункты меню 1.

{

Console.WriteLine("1) Действие с двумерным массивом.");

Console.WriteLine("2) Действие со строкой.");

Console.WriteLine("3) Выход");

}

static void Commands2() //пункты меню 1.1.

{

Console.WriteLine("1) Создать двумерный массив вручную");

Console.WriteLine("2) Создать двумерный массив при помощи ДСЧ");

Console.WriteLine("3) Напечатать двумерный массив");

Console.WriteLine("4) Удалить из двумерного массива первую строку, в которой есть хотя бы один элемент равный 0");

Console.WriteLine("5) Назад");

}

static void Commands3() //пункты меню 1.2.

{

Console.WriteLine("1) Ввести текст вручную");

Console.WriteLine("2) Сгенерировать строку из массива заранее сформированных тестовых строк");

Console.WriteLine("3) Напечатать текст");

Console.WriteLine("4) Перевернуть каждое четное слово");

Console.WriteLine("5) Назад");

}

static void DeleteFirstStrWith0(ref double[,] matr) //удаление первой строки с 0

{

int ind = -1;

for (int i = 0; i < matr.GetLength(0); i++)

{

for (int j = 0; j < matr.GetLength(1); j++)

{

if (matr[i, j] == 0 && ind == -1)

{

ind = i;

}

}

}

if (ind != -1)

{

double[,] temp = new double[matr.GetLength(0) - 1, matr.GetLength(1)];

for (int i = 0; i < matr.GetLength(0); i++)

{

for(int j = 0; j < temp.GetLength(1); j++)

{

if (i < ind)

{

temp[i,j] = matr[i, j];

}

else if (i > ind)

{

temp[i-1,j] = matr[i, j];

}

}

}

matr = temp;

Console.WriteLine("Первая строка с 0 удалена.");

}

else Console.WriteLine("В двумерном массиве нет строки с 0");

}

static string InputStr(ref string str) // ввод строки

{

Console.WriteLine("Напишите ниже свой текст:");

str = Console.ReadLine().ToString();

Console.WriteLine("Текст введён");

return str;

}

static string InputStrRnd(ref string str) // генерация случайной строки из имеющихся

{

string[] strRnd = { "Следует отметить, что глубокий уровень погружения напрямую зависит от инновационных методов управления процессами. Лишь активно развивающиеся страны третьего мира, которые представляют собой яркий пример континентально - европейского типа политической культуры, будут объявлены нарушающими общечеловеческие нормы этики и морали.", "А также независимые государства представляют собой не что иное, как квинтэссенцию победы маркетинга над разумом и должны быть преданы социально - демократической анафеме.", "Внезапно, независимые государства представляют собой не что иное, как квинтэссенцию победы маркетинга над разумом и должны быть разоблачены.", "Ясность нашей позиции очевидна: высокое качество позиционных исследований напрямую зависит от поставленных обществом задач." };

Random rnd = new Random();

str = strRnd[rnd.Next(strRnd.Length)];

Console.WriteLine("Выбран случайный текст из имеющегося списка");

return str;

}

static void ReverseEvenWord(ref string str) // переворот каждого чётного слова

{

if (string.IsNullOrWhiteSpace(str) == true) Console.WriteLine("Текста нет(");

else

{

str = str.Trim(); //убрать пробелы в начале и в конце предложения, если они есть

string[] sent = str.Split(' ');

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

{

if (i % 2 == 1)

{

string t = sent[i];

char[] chars = t.ToCharArray();

if (t.All(c => char.IsLetterOrDigit(c))) // проверка последнего символа слова

{

Array.Reverse(chars);

sent[i] = new string(chars);

}

else

{

string sign = t[t.Length - 1].ToString(); // сохранение символа, если он есть, в отдельную переменную

t = t.Substring(0, t.Length - 1);

chars = t.ToCharArray();

Array.Reverse(chars);

sent[i] = new string(chars) + sign;

}

}

}

string msg = ""; // восстанавливаем строку из массива

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

{

msg += sent[i] + " ";

}

str = msg;

Console.WriteLine("Чётные слова перевёрнуты");

}

}

static void PrintStr(ref string str) //печать строки

{

if (string.IsNullOrWhiteSpace(str) == true) Console.WriteLine("Текста нет(");

else

{

Console.WriteLine("Ваш текст:");

Console.WriteLine(str);

}

}

static void Main(string[] args)

{

int ans = 0;

do

{

Commands1();

ans = InputIntNumber();

switch (ans)

{

case 1:

{

double[,] matr = new double[0, 0];

do

{

Commands2();

ans = InputIntNumber();

switch (ans)

{

case 1:

{

CreateMatr(ref matr);

break;

}

case 2:

{

CreateMatrRnd(ref matr);

break;

}

case 3:

{

PrintMatr(matr);

break;

}

case 4:

{

DeleteFirstStrWith0(ref matr);

break;

}

}

} while (ans != 5);

break;

}

case 2:

{

string str = "";

do

{

Commands3();

ans = InputIntNumber();

switch (ans)

{

case 1:

{

InputStr(ref str);

break;

}

case 2:

{

InputStrRnd(ref str);

break;

}

case 3:

{

PrintStr(ref str);

break;

}

case 4:

{

ReverseEvenWord(ref str);

break;

}

}

} while (ans != 5);

break;

}

}

} while (ans !=3);

}

}

}

## 3.3. Тестирование.

### 3.3.1. Ввод строки.

Результаты тестирования представлены на рисунке 3.3.1. По таблице видно, что все ожидаемые результаты совпадают с реальными.

|  |  |  |  |
| --- | --- | --- | --- |
| **Номер теста** | **Переменная** | **Ожидаемый результат** | **Реальный результат** |
| 1 | В лесу родилась елочка. В лесу она росла. Зимой и летом стройная, зеленая была. | Текст введён | Текст введён |
| 2 | 12312 23123 62 | Текст введён | Текст введён |
| 3 | заащ 22щщ 2 | Текст введён | Текст введён |
| 4 |  | Текст введён | Текст введён |

***Таблица 3.3.1. – Тесты функции 1***

### 3.3.2. Генерация случайной строки из имеющихся.

Результаты тестирования представлены на рисунке 3.3.2. По таблице видно, что все ожидаемые результаты совпадают с реальными. Для этой функции был произведён всего 1 текст, так как ничего вводить не требуется.

|  |  |  |  |
| --- | --- | --- | --- |
| **Номер теста** | **Переменная** | **Ожидаемый результат** | **Реальный результат** |
| 1 |  | * Один элемент из массива имеющихся * Сообщение (Выбран случайный текст из имеющегося списка) | * Один элемент из массива имеющихся * Сообщение (Выбран случайный текст из имеющегося списка) |

***Таблица 3.3.2. – Тесты функции 2***

### 3.3.3. Переворот каждого чётного слова.

Результаты тестирования представлены на рисунке 3.3.3. По таблице видно, что все ожидаемые результаты совпадают с реальными.

| **Номер теста** | **Переменная** | **Ожидаемый результат** | **Реальный результат** |
| --- | --- | --- | --- |
| 1 | В лесу родилась елочка. В лесу она росла. Зимой и летом стройная, зеленая была. | В усел родилась акчоле. В усел она алсор. Зимой и летом яанйортс, зеленая алыб. | В усел родилась акчоле. В усел она алсор. Зимой и летом яанйортс, зеленая алыб. |
| 2 | карандаш чёрный, об этом. | карандаш йынрёч, об мотэ. | карандаш йынрёч, об мотэ. |
| 3 | карандаш йынрёч, об мотэ. | карандаш чёрный, об этом. | карандаш чёрный, об этом. |
| 4 | 112 9139021, efjef389893289 / . | 112 1209319, efjef389893289 / . | 112 1209319, efjef389893289 / . |
| 5 |  | Текста нет | Текста нет |

***Таблица 3.3.3. – Тесты функции 3***

### 3.3.4. Печать строки.

Результаты тестирования представлены на рисунке 3.3.4. По таблице видно, что все ожидаемые результаты совпадают с реальными.

|  |  |  |  |
| --- | --- | --- | --- |
| **Номер теста** | **Переменная** | **Ожидаемый результат** | **Реальный результат** |
| 1 | В лесу родилась елочка. В лесу она росла. Зимой и летом стройная, зеленая была. | В лесу родилась елочка. В лесу она росла. Зимой и летом стройная, зеленая была. | В лесу родилась елочка. В лесу она росла. Зимой и летом стройная, зеленая была. |
| 2 | 123123 1231233 | 123123 1231233 | 123123 1231233 |
| 3 |  | Текста нет | Текста нет |

***Таблица 3.3.4. – Тесты функции 4***