Лабораторная работа №7:

№1:

using System;

using static System.Console;

namespace Task01

{

class Program

{

static void Main(string[] args)

{

WriteLine("Введите текст, с которым будет производиться работа:");

var text = ReadLine();

Write("Введите символ, который нужно найти: ");

char character = char.Parse(ReadLine());

double percentage = EntryPercentage(text, character);

WriteLine("\n" + ConvertToMessage(percentage));

}

static double EntryPercentage(string text, char character)

{

int entryCount = 0;

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

{

if (text[i] == character)

{

entryCount++;

}

}

double percentage = (double) entryCount / text.Length;

return percentage;

}

static string ConvertToMessage(double percentage)

{

if (percentage == 0)

{

return "\* Символ не найден \*";

}

return $"Символ состявляет {(percentage \* 100):N2}% от всего текста";

}

}

}

№2:

using System;

using static System.Console;

namespace Task02

{

class Program

{

static void Main(string[] args)

{

WriteLine("Введите текст, с которым будет производиться работа:");

var text = ReadLine();

var encodedText = ReverseWordSpelling(text);

WriteLine("\nЗашифрованный текст:");

WriteLine(encodedText);

var decodedText = ReverseWordSpelling(encodedText); ;

WriteLine("\nРасшифрованный текст:");

WriteLine(decodedText);

}

static bool IsWordCharacter(char character)

{

var isEnglishCharacter =

(character >= 'a' && character <= 'z')

|| (character >= 'A' && character <= 'Z');

var isRussianCharacter =

(character >= 'а' && character <= 'я')

|| (character >= 'А' && character <= 'Я')

|| (character == 'ё') || (character == 'Ё');

return isEnglishCharacter || isRussianCharacter;

}

static string ReverseWordSpelling(string text)

{

var characters = text.ToCharArray();

int startIndex = 0;

int endIndex = 0;

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

{

if (!IsWordCharacter(characters[i]))

{

endIndex = i;

int length = endIndex - startIndex;

for (int k = 0; k < length / 2; k++)

{

char temp = characters[startIndex + k];

characters[startIndex + k] = characters[endIndex - 1 - k];

characters[endIndex - 1 - k] = temp;

}

startIndex = endIndex + 1;

}

}

return new string(characters);

}

}

}

№3:

using System;

using static System.Console;

namespace Task03

{

class Program

{

static void Main(string[] args)

{

WriteLine("Введите текст, с которым будет производиться работа: ");

var text = ReadLine();

WriteLine("Текст, поделённый на подстроки");

SplitIntoSubstrings(text, 10);

}

static void SplitIntoSubstrings(string text, int maxLength)

{

var allWords = text.Split(' ');

int length = 0;

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

{

if (length + allWords[i].Length < maxLength)

{

if (length > 0)

{

Write(' ');

}

length += allWords[i].Length;

Write(allWords[i]);

}

else

{

Write('\n');

length = 0;

}

}

}

}

}