1. **Строки. Классы String и StringBuilder**

Задание 1. Вывести только те слова сообщения, которые содержат хотя бы одну цифру.

Листинг программы:

namespace Task\_1

{

class Program

{

public static void Main(string[] args)

{

Console.Write("Введите предложение: ");

string text = Console.ReadLine();

Console.WriteLine(GetMassivWordsInsertNumber(AddWordsInMassiv(text)));

}

public static string[] AddWordsInMassiv(string text)

{

string[] words = text.Split(new char[] { ' ' }, StringSplitOptions.RemoveEmptyEntries);

return words;

}

public static string GetMassivWordsInsertNumber(string[] words)

{

string text = "";

foreach (string word in words)

{

if (word.LastIndexOfAny(new char[] { '1', '2' , '3', '4', '5' , '6', '7', '8', '9', '0' }) > -1)

{

text += " " + word;

}

}

return text;

}

}

}

Таблица 1.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Варя6 приш3л на парус посмотреть 1и1 | Варя6 приш3л 1и1 |

Анализ результатов:



Рисунок 1.1 – Результат работы программы

Источник: собственная разработка

Задание 2. Составить программу, которая будет вводить строку в переменную string. Найти в ней те слова, которые начинаются и оканчиваются одной и той же буквой.

Листинг программы:

namespace Task\_2

{

class Program

{

public static void Main(string[] args)

{

Console.Write("Введите предложение: ");

string text = Console.ReadLine().ToLower();

Console.WriteLine(GetTextWhereFerstEquallyLast(AddWordsInMassiv(text)));

}

public static string[] AddWordsInMassiv(string text)

{

string[] words = text.Split(new char[] { ' ' }, StringSplitOptions.RemoveEmptyEntries);

return words;

}

public static string GetTextWhereFerstEquallyLast(string[] words)

{

string text = "";

foreach (string word in words)

{

if (word.LastIndexOf(word[0]) == word.Length - 1)

{

text += " " + word;

}

}

return text;

}

}

}

Таблица 2.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Поп пришел кушать и оно выпало из ада | Поп и оно ада |

Анализ результатов:



Рисунок 2.1 – Результат работы программы

Источник: собственная разработка

Задание 3. С клавиатуры вводится предложение. Результаты всех действий вывести на экран.

Поменять местами первое и последнее слова в предложении.

Склеить второе и третье слова в предложении.

Третье слово предложения вывести в обратном порядке

В первом слове предложения вырезать первые две буквы.

Листинг программы:

namespace Task\_3

{

class Program

{

public static void Main(string[] args)

{

Console.Write("Введите предложение: ");

string text = Console.ReadLine();

Console.WriteLine($"\nПоменять местами первое и последнее слова в предложении \n{GetSubstitution(text)}");

Console.WriteLine($"\nСклеить второе и третье слова в предложении \n{GetGluiSecondAndThirdWords(text)}");

Console.WriteLine($"\nТретье слово предложения вывести в обратном порядке \n{GetReversThirdWord(text)}");

Console.WriteLine($"\nВырезать первые две буквы \n{CutTwoCharInFerstWord(text)}");

}

public static string GetSubstitution(string text)

{

string ferstWord = text.Substring(0, text.IndexOf(" "));

string lastWord = text.Substring(text.LastIndexOf(" ") + 1);

string newText = text.Remove(text.LastIndexOf(" ") + 1);

return newText.Replace(ferstWord, lastWord) + ferstWord;

}

public static string GetGluiSecondAndThirdWords(string text)

{

return text.Remove(text.IndexOf(" ", text.IndexOf(" ") + 1), 1);

}

public static string GetReversThirdWord(string text)

{

string word = text.Substring(text.IndexOf(" ") + 1, text.IndexOf(" "));

string reversWord = "";

for (int i = word.Length - 1; i >= 0; i--)

{

reversWord += word[i];

}

return reversWord;

}

public static string CutTwoCharInFerstWord(string text)

{

return text.Remove(0, 2);

}

}

}

Таблица 3.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Верхушка айсберга видна сегодня |  |

Анализ результатов:

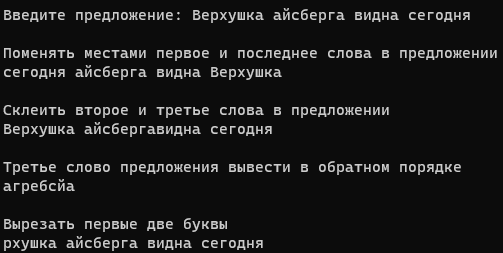


Рисунок 3.1 – Результат работы программы

Источник: собственная разработка

Задание 4. Известны фамилия, имя и отчество пользователя. Найти его код личности. Правило получения кода личности: каждой букве ставится в соответствие число - порядковый номер буквы в алфавите. Эти числа складываются. Если полученная сумма не является однозначным числом, то цифры числа снова складываются и так до тех пор, пока не будет получено однозначное число.

Листинг программы:

namespace Task\_4

{

class Program

{

private static void Main()

{

Console.Write("Введите ваше ФИО: ");

string name = Console.ReadLine().ToLower();

int code = GetCode(GetCalculateSum(name));

Console.WriteLine($"Код: {code}");

}

public static int GetCode(int number)

{

int code = 0;

while (number >= 1)

{

code += number % 10;

number /= 10;

}

return code;

}

private static int GetCalculateSum(string fio)

{

char[] alphavite = " абвгдеёзжийклмнопрстуфхцчшщъыьэюя".ToCharArray();

int sum = 0;

foreach (char s in fio)

{

for (int i = 0; i < 32; i++)

{

if (s == alphavite[i])

{

sum += i;

}

}

}

return sum;

}

}

}

Таблица 4.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Микитчук Вячеслав Евгеньевич | 5 |

Анализ результатов:



Рисунок 4.1 – Результат работы программы

Источник: собственная разработка