**6 СТРОКИ. КЛАССЫ STRING И STRINGBUILDER**

Задание 1. Дан фрагмент текста, запрашиваемый у пользователя. Написать программу, находящую наибольшее количество цифр, идущих в нем подряд.

Листинг программы:

try

{

Console.Write("Введите строку: ");

string input = Console.ReadLine();

int carent = default;

int max = default;

int count = 0;

int maxChar = 0;

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

{

if (char.IsDigit(input[i]))

{

carent += 1;

}

else if (carent > max)

{

max = carent;

carent = 0;

}

}

if (carent > max)

{

max = carent;

}

Console.WriteLine(max);

Console.ReadLine();

}

catch (Exception ex)

{

Console.WriteLine(ex.Message);

}

Таблица 6.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 123a1234 | 4 |

Анализ результатов:



Рисунок 6.1 – Результаты работы программы

Источник: собственная разработка

Задание 2 Дано предложение. Определить количество букв «a», предшествующих первой запятой предложения. Рассмотреть два случая: 1) известно, что запятые в предложении есть; 2) запятых в предложении может не быть.

Листинг программы:

try

{

Console.Write("Введите строку: ");

string input = Console.ReadLine();

var pos = input.IndexOf(',');

int count = 0;

if (pos > 0)

{

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

{

if (input[i] == 'a')

{

count += 1;

Console.WriteLine(input[i]);

}

}

}

else

{

Console.WriteLine("Запятые отсутствуют");

}

Console.WriteLine(count);

}

catch (Exception ex)

{

Console.WriteLine(ex.Message);

}

Таблица 6.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| data,123 | a  a  2 |

Анализ результатов:

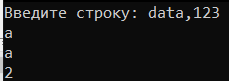


Рисунок 6.2 – Результаты работы программы

Источник: собственная разработка