

ПРАКТИЧЕСКАЯ РАБОТА № 9

(4 академических часа)

Тема: работа со строками в C#.

Цель задания:

Получение практических навыков по составлению алгоритмов и программ обработки данных строкового типа.

Методика выполнения работы.

В языке C# строковые значения представляет тип `string`, а вся функциональность работы с данным типом сосредоточена в классе `System.String`.

Создавать строки можно, как используя переменную типа `string` и присваивая ей значение, так и применяя один из конструкторов класса `String`:

```
string s1 = "hello";  
string s2 = null;
```

```
string s3 = new String('a', 6); // результатом будет строка "aaaaaa"  
string s4 = new String(new char[]{'w', 'o', 'r', 'l', 'd'});
```

Класс `string` обладает богатым набором методов для сравнения строк, поиска в строке и других действий со строками:

| Название | Вид | Описание |
|---|---------------------|---|
| Compare | Статический метод | Сравнение двух строк в лексикографическом (алфавитном) порядке. Разные реализации метода позволяют сравнивать строки с учетом или без учета регистра. |
| CompareTo | Метод | Сравнение текущего экземпляра строки с другой строкой. |
| Concat | Статический метод | Слияние произвольного числа строк. |
| Copy | Статический метод | Создание копии строки |
| Empty | Статическое поле | Открытое статическое поле, представляющее пустую строку |
| Format | Статический метод | Форматирование строки в соответствии с заданным форматом |
| IndexOf, IndexOfAny, LastIndexOf, LastIndexOfAny | Экземплярные методы | Определение индексов первого и последнего вхождения заданной подстроки или любого символа из заданного набора в данную строку. |
| Insert | Экземплярный метод | Вставка подстроки в заданную позицию |
| Join | Статический метод | Слияние массива строк в единую строку. Между элементами массива вставляются разделители. |
| Length | Свойство | Возвращает длину строки |

| | | |
|---------------------------------|---------------------|---|
| PadLeft, PadRighth | Экземплярные методы | Выравнивают строки по левому или правому краю путем вставки нужного числа пробелов в начале или в конце строки. |
| Remove | Экземплярный метод | Удаление подстроки из заданной позиции |
| Replace | Экземплярный метод | Замена всех вхождений заданной подстроки или символа новыми подстрокой или символом. |
| Split | Экземплярный метод | Разделяет строку на элементы, используя разные разделители. Результаты помещаются в массив строк. |
| StartWith, EndWith | Экземплярные методы | Возвращают true или false в зависимости от того, начинается или заканчивается строка заданной подстрокой. |
| Substring | Экземплярный метод | Выделение подстроки, начиная с заданной позиции |
| ToCharArray | Экземплярный метод | |
| ToLower, ToUpper | Экземплярные методы | |
| Trim, TrimStart, TrimEnd | Экземплярные методы | |

Примеры:

1. Соединение строк

```
string str1 = "Первая строка";
string str2 = string.Copy(str1);
string str3 = "Вторая строка";
string str4 = "ВТОРАЯ строка";
string strUp, strLow;
int result, idx;
Console.WriteLine("str1: " + str1);
Console.WriteLine("Длина строки str1: " + str1.Length);
```

2. Создание прописной и строчной версии строки str1

```
strLow = str1.ToLower();
strUp = str1.ToUpper();
Console.WriteLine("Строчная версия строки str1: "
+ strLow);
Console.WriteLine("Прописная версия строки str1: " + strUp);
Console.WriteLine();
```

3. Сравнение строк

```
result = str1.CompareTo(str3);
if (result == 0) Console.WriteLine("str1 и str3 равны.");
else if (result < 0) Console.WriteLine("str1 меньше, чем str3");
else Console.WriteLine("str1 больше, чем str3");
Console.WriteLine();
```

4. Сравнение частей строк

```
result = String.Compare(str1, 4, str2, 4, 2);
if (result == 0) Console.WriteLine("часть str1 и str2 равны");
```

```
else Console.WriteLine("часть str1 и str2 не равны");  
Console.WriteLine();
```

5. Поиск строк.

```
idx = str2.IndexOf("строка");  
Console.WriteLine("Индекс первого вхождения подстроки строка: " + idx);  
idx = str2.LastIndexOf("o");  
Console.WriteLine("Индекс последнего вхождения символа o: " + idx);
```

6. Конкатенация

```
string str=String.Concat(str1, str2, str3, str4);  
Console.WriteLine(str);
```

7. Удаление подстроки

```
str=str.Remove(0,str1.Length);  
Console.WriteLine(str);
```

8. Замена подстроки "строка" на пустую подстроку

```
str=str.Replace("строка", "");  
Console.WriteLine(str);
```

Для каждого задания написать 2 программы: без использования методов класса и с использованием методов класса.

Задание 1. Обработка строк. Поиск и замена

1. С клавиатуры вводится строка. Подсчитать количество содержащихся в ней цифр.
2. С клавиатуры вводится строка. Если она представляет собой запись целого числа, то вывести 1, если вещественного (с дробной частью) — вывести 2; если строку нельзя преобразовать в число, то вывести 0. Считать, что дробная часть вещественного числа отделяется от его целой части десятичной точкой «.».
3. С клавиатуры вводится целое положительное число. Вывести символы, изображающие цифры этого числа (в порядке слева направо).
4. С клавиатуры вводится строка, изображающая арифметическое выражение вида « \pm . . \pm », где на месте знака операции « \pm » находится символ «+» или «-» (например, «4+7-2-8»). Вывести значение данного выражения (целое число).
5. С клавиатуры вводятся положительные числа N1 и N2 и строки S1 и S2. Получить из этих строк новую строку, содержащую первые N1 символов строки S1 и последние N2 символов строки S2 (в указанном порядке).
6. С клавиатуры вводится строка. В заданном тексте везде букву "a" замените на букву "б", а букву "б" — на букву "a".
7. С клавиатуры вводится строка, изображающая целое положительное число. Вывести сумму цифр этого числа.
8. С клавиатуры вводится строка. Вывести строку, содержащую те же символы, но расположенные в обратном порядке.
9. С клавиатуры вводится строка S. Вывести строку, содержащую символы строки S, между которыми вставлено по одному пробелу.
10. С клавиатуры вводится строка S, состоящая из нулей и единиц. Подсчитать количество нулей и единиц.
11. С клавиатуры вводятся строки S и S1. Удалить из строки S первую подстроку, совпадающую с S1. Если совпадающих подстрок нет, то вывести строку S без изменений.

12. С клавиатуры вводится строка S. Найти количество знаков препинания в этой строке.
13. С клавиатуры вводится строка S. Вывести на экран только русские буквы, содержащиеся в этой строке.
14. С клавиатуры вводится строка S (строка содержит арифметическое выражение). Если в строке соблюдается баланс открывающихся и закрывающихся круглых скобок, вывести сообщение «Выражение правильное», иначе - сообщение «Выражение неправильное».

Задание 2. Анализ и преобразование слов в строке

1. С клавиатуры вводится строка, состоящая из слов, разделённых одним или несколькими пробелами. Удалить повторные вхождения каждого слова.
2. С клавиатуры вводится строка, состоящая из N символов (N вводится с клавиатуры). Поменять местами первый и последний символы. Удалить из полученного текста все пробелы.
3. С клавиатуры вводится символ C и строки S, S1. Перед каждым вхождением символа C в строку S вставить строку S1.
4. С клавиатуры вводится строка S, состоящая из слов, разделённых одним или несколькими пробелами. Упорядочить слова предложения в алфавитном порядке.
5. С клавиатуры вводятся строки S, S1. Удалить из строки S первую подстроку, совпадающую с S1. Если совпадающих подстрок нет, то вывести строку S без изменений.
6. С клавиатуры вводятся строки S, S1. Удалить из строки S все подстроки, совпадающие с S1. Если совпадающих подстрок нет, то вывести строку S без изменений.
7. С клавиатуры вводится строка S, состоящая из слов, разделённых одним или несколькими пробелами. Найти самое короткое и самое длинное слово.
8. С клавиатуры вводятся строки S, S1, S2. Заменить в строке S все вхождения строки S1 на строку S2.
9. С клавиатуры вводится строка S, содержащая по крайней мере один символ пробела. Вывести подстроку, расположенную между первым и последним пробелом исходной строки. Если строка содержит только один пробел, то вывести пустую строку.
10. С клавиатуры вводится строка S, содержащая полное имя файла. Выделить из этой строки название первого каталога (без символов «\»). Если файл содержится в корневом каталоге, то вывести символ «\».
11. С клавиатуры вводится строка S. Зашифровать ее, поместив вначале все символы, расположенные на четных позициях строки, а затем, в обратном порядке, все символы, расположенные на нечетных позициях (например, строка «Программа» превратится в «ргамамроП»).
12. С клавиатуры вводятся строки S, S1. Каждый встречающийся в строке заданный символ A (вводится с клавиатуры) заменить на заданную последовательность символов S1, расширяя при этом строку.
13. С клавиатуры вводится строка S, состоящая из слов, разделённых одним или несколькими пробелами. Вывести на экран все слова, преобразовав каждое при этом следующим образом: первую букву слова заменить последней.
14. С клавиатуры вводится строка S, состоящая из слов, разделённых одним или несколькими пробелами. Преобразовать ее так, чтобы между словами был ровно один пробел.

Контрольные вопросы.

1. Какими способами можно создать строку?
2. К каким типам данных относятся строки?
3. Какова максимально возможная длина строки?

4. С величиной какого типа данных совместим по присваиванию отдельный символ строки?
5. Как осуществляется доступ к отдельному символу строки?
6. Какая функция (процедура) является аналогом операции сцепления (+) при работе со строками?