Программирование на языке Си

Тема 2.Организация ввода-вывода данных. Форматирование.

```
// На экран выводится значение идентификатора x
Console.WriteLine(x);
```

Пример:

```
x=5;
Console.WriteLine(x);
```

```
/* На экран выводится строка, образованная последовательным
слиянием строки "x=", значения x, строки "y=" и значения y */
Console.WriteLine("x=" + x +"y=" + y);
```

Пример:

```
x=5;
y=6;
Console.WriteLine(“x=“+x+”y=“+y);
```

```
x=5y=6
```

Пример :

```
Console.Write(“a=“);
a =Convert.ToInt32(Console.ReadLine());
```

```
a=
```

Использование управляющих последовательностей

Управляющей последовательностью называют определенный символ, предваряемый обратной косой чертой.

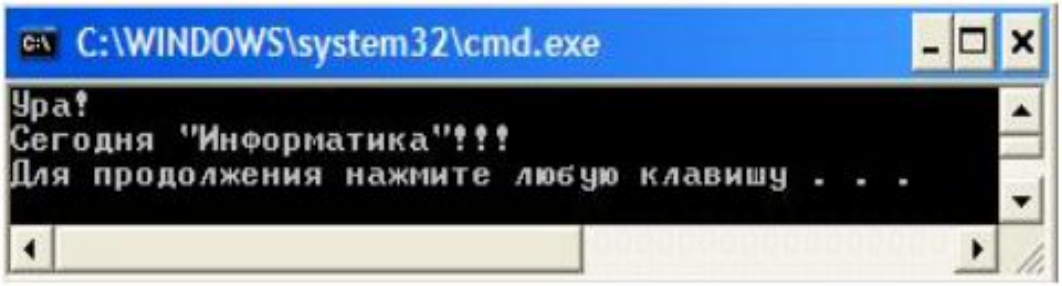
Используется для представления кодов символов , не имеющих графического представления (перевод курсора) или символов, имеющих специальное обозначение.

Управляющие символы

Вид	Наименование
\a	Звуковой сигнал
\b	Возврат на шаг назад
\f	Перевод страницы
\n	Перевод строки
\r	Возврат каретки
\t	Горизонтальная табуляция
\v	Вертикальная табуляция
\\	Обратная косая черта
\'	Апостроф
\"	Кавычки

Пример:

```
static void Main()
{
  Console.WriteLine("Ура!\nСегодня \"Информатика\"!!!");
}
```



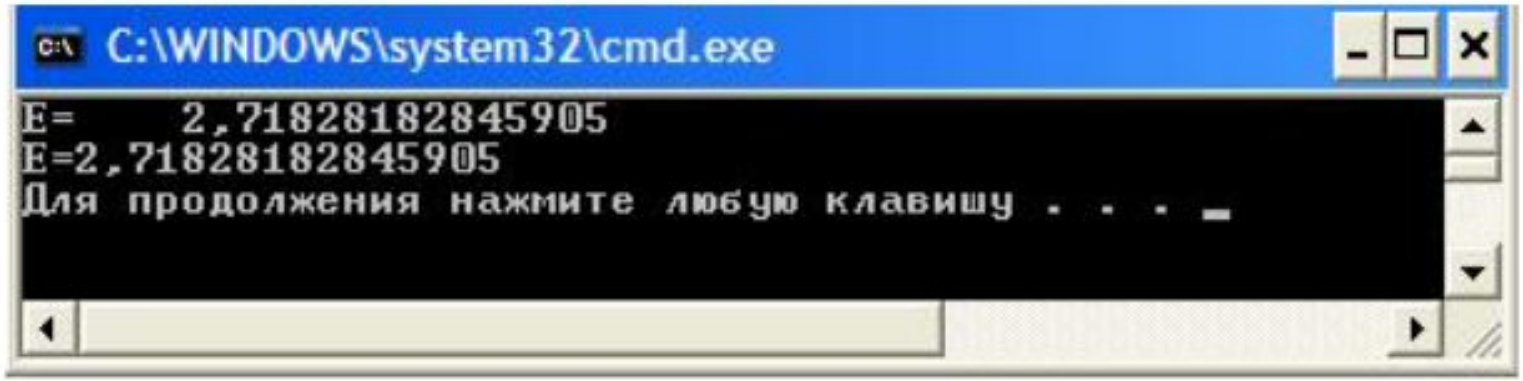
Замечание. Вместо управляющей последовательности \n можно использовать константу Enviropment.NewLine. Она более универсальна, так как ее значение зависит от контекста и операционной системы, в которой запускается программа.

Задание. Измените программу так, чтобы все сообщение выводилось в одну строку, а после вывода сообщения раздавался звуковой сигнал.

Управление размером поля вывода

Первым аргументом указывается строка вида {n,m} - где n определяет номер идентификатора из списка, а m- количество позиций (размер поля вывода), отводимых под значение данного идентификатора.

```
static void Main()
{
   double x = Math.E;
   Console.WriteLine("E={0,20}", x);
   Console.WriteLine("E={0,10}", x);
}
```

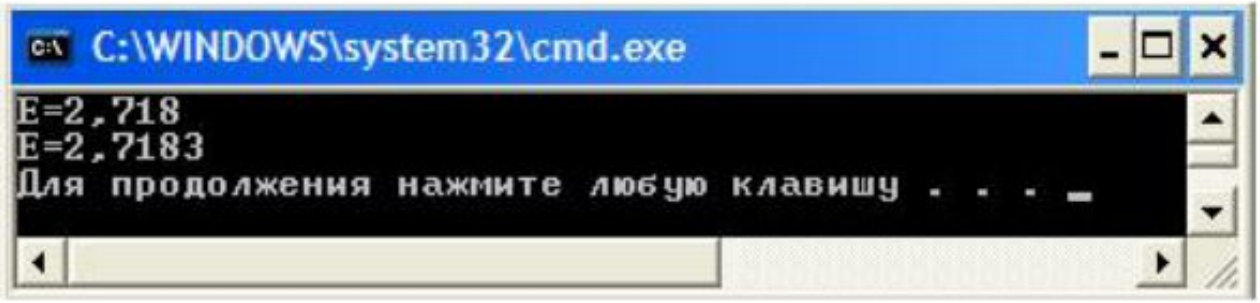


Управление размещением вещественных чисел

Первым аргументом указывается строка вида {n:##.###} – где n определяет номер идентификатора, а ##.### определяет формат вывода вещественного числа.

Целая часть две позиции, дробная – три.

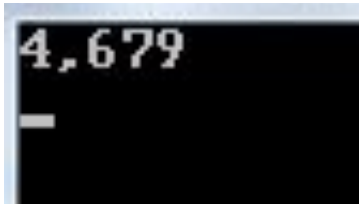
```
static void Main()
{
   double x= Math.E;
   Console.WriteLine("E={0:##.###}", x);
   Console.WriteLine("E={0:.####}", x);
}
```



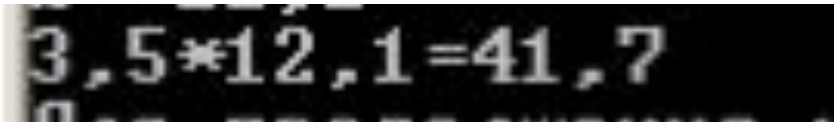
Задание. Измените программу так, чтобы число e выводилось на экран с точностью до 6 знаков после запятой.

Примеры:

```
a = 4.678999;
Console.WriteLine("{0:##.###}", a);
```



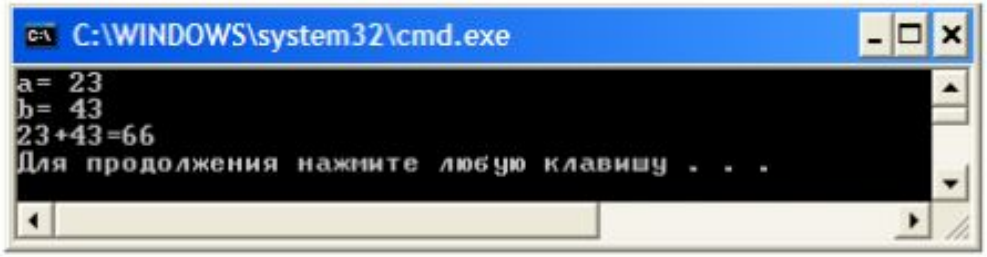

```
Console.Write("{0:##.#}", a);
Console.Write(“ * ”);
Console.Write("{0:##.#}", b);

Console.Write(“=“ + c);
```



Практикум

I. Написать программу, которая реализует диалог с пользователем:

1) запрашивает с клавиатуры два целых числа и выводит на экран сумму данных чисел:




```
Int a,b,c;
Console.Write(“a=“);
a =Convert.ToInt32(Console.ReadLine());
……..
c=a+b;
Console.WriteLine(a+“+“+b+”=“+c);
```

Практикум _2 к лекции 3