

Sistemas de Numeración

Un sistema de numeración está definido por una secuencia (ordenada) de n símbolos s_0, s_1, \dots, s_{n-1} a los cuales se les asocia los valores de 0 a $n-1$ respectivamente. El sistema decimal (de base diez), que es el que más comúnmente utilizamos, está constituido por diez símbolos: '0', '1', '2', '3', '4', '5', '6', '7', '8', y '9' que corresponden en este caso a los valores enteros 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 respectivamente. En este sistema la secuencia 475 expresa al número cuatrocientos setenta y cinco, resultante de multiplicar 4 por 10^2 + 7 por 10^1 + 5 por 10^0 .

Si el sistema estuviese constituido por los tres símbolos 'a', 'b', 'c' (que se asocian a los valores enteros 0, 1 y 2 respectivamente) entonces la secuencia *baac* denotaría al número cuyo valor es $1 * 3^3 + 0 * 3^2 + 0 * 3^1 + 2 * 3^0$, es decir, el 29 en el sistema decimal.

En general, en un sistema con n símbolos s_0, s_1, \dots, s_{n-1} , si un símbolo s_i ocupa la k -ésima posición de derecha a izquierda en un número, entonces aporta una cantidad $i * n^{k-1}$ al valor total del número.

Implemente una biblioteca de clases de nombre `SistemasNumeracion` (al compilarse debe producir un ensamblado `SistemasNumeracion.dll`) con el siguiente método:

```
namespace SistemasNumeracion
{
    public class Examen
    {
        public static int ParseInt(string numero, char[] digitosBase)
        {
            // aquí va su código...
        }
    }
}
```

que a partir de una secuencia expresada por el `string` `numero` y la base representada por el `array` `digitosBase`, devuelva el valor de tipo `int` correspondiente a ese número según la base. Los caracteres que constituyen los dígitos de la base aparecerán en el `array` `digitosBase` en orden creciente de valor: el primero es el que vale 0 y el último es el de máximo valor.

De acuerdo con el ejemplo anterior, si se invoca:

```
ParseInt("baac", new char[] { 'a', 'b', 'c' })
```

se debe devolver el valor entero 29.

Aclaraciones:

- Al evaluar su método, la cadena `numero` siempre va a ser distinta de `null`, de longitud mayor o igual a 1 y todos sus caracteres pertenecerán al `array` `digitosBase`.
- Se garantiza que el `array` `digitosBase` siempre será no nulo, de longitud mayor o igual que 2 y no contendrá caracteres repetidos.

(este enunciado tiene 2 (dos) páginas)

- En el caso de los caracteres alfabéticos se hará distinción entre mayúsculas y minúsculas, es decir, el carácter 'x' es totalmente distinto del 'X' y pueden aparecer ambos en una misma base, en un mismo número, etc.