
Tipos simples y strings (2)

X26700_es

Se piden los códigos de varias funciones. En cada caso bastan unas pocas líneas de código. Seguid el formato de los ejemplos que se muestran abajo. *Atención:* No uséis el método `split` de la clase `string`.

1. Programad una función `um_count(s)` que dado un string `s` retorne el número de veces que la letra `u` es seguida por la letra `m` en el string `s`.
2. Programad una función entera `word_count(s)` that retorne el número de palabras en el string `s`. Se supone que todos los caracteres de `s` son letras o espacios.
3. Programad una función `kth_word(s, k)` que dados un string `s` y un entero $k \geq 1$ retorne la palabra que ocupa la *k-ésima* posición en `s`. Si `s` tiene menos de `k` palabras la función debe devolver la palabra vacía. Se supone que todos los caracteres de `s` son letras o espacios.
4. Programad una función `suc_word(s)` que dado un string `s` retorna la primera palabra en `s` que tiene alguna letra mayúscula. Si todas las letras en `s` son minúsculas retorna el string vacío. Se supone que todos los caracteres de `s` son letras o espacios.
5. Programad una función `drawA(n)` que dado un entero impar $n \geq 3$ escribe la letra `A` de tamaño `n` formada con el símbolo `*`.

Puntuación

Cada función vale 20 puntos.

Ejemplo de sesión

```
>>> um_count("Qui invenit amicum invenit thesauruM")
1
>>> word_count("Alea iacta          est")
3
>>> kth_word("Alea iacta est", 3)
est
>>> suc_word("qui invenit amiCum invenit thesauruM")
amiCum
>>> drawA(5)
  *
 * *
*****
 *   *
*     *
```

Información del problema

Autor : Jorge Castro

Generación : 2016-09-14 10:49:54

© Jutge.org, 2006–2016.

<http://www.jutge.org>