Homework 2015

Juana Paulina Águila Hernández

Características del equipo en donde se hicieron las pruebas:

* **Sistema operativo:** OS X Yosemite (10.10.4)
* **Procesador:** 2.5 GHz Intel Core i5
* **Memoria:** 4 GB 1600 MHz DDR3
* **Versión de Java:** 1.6.0\_65-b14-466.1-11M4716
* **Memoria disponible:** 81 MB (Información proporcionada por la clase *Runtime*).
* **Memoria máxima disponible:** 123 MB (Información proporcionada por la clase *Runtime*).

Archivo donde se ejecutaron las pruebas: package/Principal/TestZone.java

Repositorio en github: https://github.com/vegenisennawa/Homework2015.git

Ejercicios:

1. Implementar la siguiente recursión:
   1. Dado un elemento de la clase *String* de Java, se necesita codificar una función recursiva que cuente todos los caracteres en un objeto *String*.

Este ejercicio se puede ver dentro de la clase RecSring [package/MyString/RecString.java]

Se tomó en cuenta un índice que se ingresa a la función *StrLen* y la cadena a medir.

Si el iterador ingresado es menor a la longitud de la cadena, entonces va llamando a la función recursivamente hasta que no se cumpla esa condición, regresando el nuevo iterador.

Si el iterador es mayor a la longitud de la cadena, la función regresa -1 para indicar un error.

* 1. Dé la recursión calculando T(n).

*Cálculo para cuando la cadena tiene longitud 0.*

**Código** **Pasos**

StrLen(index, cadena)

if(index < cadena.length()) 1

return *StrLen*(++index, cadena); 0

else if(index > cadena.length()) 0

index = -1; 0

return index; 1

Sumando: 1 + 1 = 2.

*Cálculo para cuando la cadena tiene longitud 1.*

**Código** **Pasos**

StrLen(index, cadena)

if(index < cadena.length()) 1

return *StrLen*(++index, cadena); 1

else if(index > cadena.length()) 0

index = -1; 0

return index; 1

Sumando: 1 + 1 + 1 = 3.

*Cálculo para cuando la cadena tiene longitud mayor a la longitud de la cadena*

**Código** **Pasos**

StrLen(index, cadena)

if(index < cadena.length()) 1

return *StrLen*(++index, cadena); 0

else if(index > cadena.length()) 1

index = -1; 1

return index; 1

Sumando: 1 + 1 + 1 +1 = 4

*Cálculo para cuando la cadena tiene longitud = 0 y longitud = n.*

**Código** **Pasos**

StrLen(index, cadena)

if(index < cadena.length()) 1

return *StrLen*(++index, cadena); n-1

else if(index > cadena.length()) 0

index = -1; 0

return index; 1

Sumando: 1 + n - 1 +1 = 1 + n

* 1. Pruebe con una muestra suficientemente grande.

1. Dos
2. Tres
3. Cuatro
4. Cinco