

75.31 Teoría de Lenguaje

75.24 Teoría de la Programación

Primer Cuatrimestre de 2015

Fecha límite de presentación: 19/06/2014

El TP se debe enviar en formato digital (pdf) y el código fuente asociado (archivos .oz) en un archivo zip a la siguiente dirección de correo electrónico: leanrafa+tdl2015@gmail.com hasta la fecha de entrega inclusive.

1. Referencias externas

Para cada una de las siguientes definiciones de procedimiento, liste las referencias externas

1. `proc {P X Y} local Z in {Q Z U} end end`
2. `proc {P X Y} local Z in {Q Z Y} end end`
3. `proc {P X Y} local Z in {P Z Y} end end`

2. Ejemplo de ejecución:

Ejecute el siguiente programa a mano usando la máquina abstracta vista en clase, y mostrando en cada paso el estado del stack y del store:

```
local X Y Z P Q in
  X=6
  Y=4
  proc {P A B}
    proc {B U V}
      V=U+A
    end
  end
  {P X Q}
  {Q Y Z}
end
```

3. Hilos

Probar el siguiente ejemplo y utilizar el panel de *OZ* para ver cuantos hilos se crean.
¿Cuántos hilos están simultaneamente en estado runnable?

```
fun {Fib N}
  case N of 0 then 0
  [] 1 then 1
  else
    thread {Fib N-1} end + thread {Fib N-2} end
  end
end
```

4. Evaluación perezosa

Considere las siguientes definiciones de una funcion para revertir una lista

```
fun lazy {Reverse1 S}
  fun {Rev S R}
    case S of nil then R
    [] X|S2 then {Rev S2 X|R} end
  end
in {Rev S nil} end
fun lazy {Reverse2 S}
  fun lazy {Rev S R}
    case S of nil then R
    [] X|S2 then {Rev S2 X|R} end
  end
in {Rev S nil} end
```

¿Cuál es la diferencia en comportamiento entre {Reverse1 [a b c]} y {Reverse2 [a b c]}?
¿Ambos llamados devuelven el mismo resultado? Explique en cada caso.

5. Parámetros

Comparar en Oz y en el lenguaje asignado en el TP qué tipos de pasaje por parámetros se soportan.

6. Mensajes

Escribir un agente que muestre en el *Browser* cada uno de los mensajes recibidos.

7. Celdas de memoria

¿Qué mostrará el siguiente programa? ¿Porqué?

```
declare
X={NewCell 0}
{Assign X 5}
Y=X
{Assign Y 10}
{Browse {Access X}==10}
{Browse X==Y}
Z={NewCell 10}
{Browse Z==Y}
{Browse @X==@Z}
```

8. Abstracción de datos

Implementar en Oz un procedimiento que reciba una lista con letras y muestre por pantalla la frecuencia de aparición de cada letra, ordenada por frecuencia y a igual cantidad de apariciones ordenadas alfabéticamente, usando como estructura de datos un diccionario con las siguientes primitivas:

- NewDicc: Crea un diccionario nuevo y vacío.
- Put: Inserta una clave y su correspondiente valor en el diccionario.
- Get: Obtiene el valor asociado a la clave ingresada.
- Domain: Retorna una lista con todas las claves del diccionario.

Siguiendo la sección 6.4 del libro, desarrollar 8 implementaciones del diccionario implementado con un árbol binario, siguiendo los ejes bundled-unbundled, opened-secure, y explicit state - declarative. Realizar una de las implementaciones en el lenguaje asignado para el TP grupal

9. Más abstracción de datos

Implementar un TDA Carta, uno Mazo de carta y otro que sea Juego (en Oz con las 8 implementaciones y una en el lenguaje propio). El juego a implementar es a elección, se sugiere uno con reglas sencillas, como ser “La casita robada” o la “Escoba de 15”.

Describe las primitivas de cada TDA y ejemplifique la utilización de una de las implementaciones.