

Trabajo Práctico 3. Programación Funcional

Puntuación

Puntaje Total: 100 puntos

Aprobación: 60 puntos

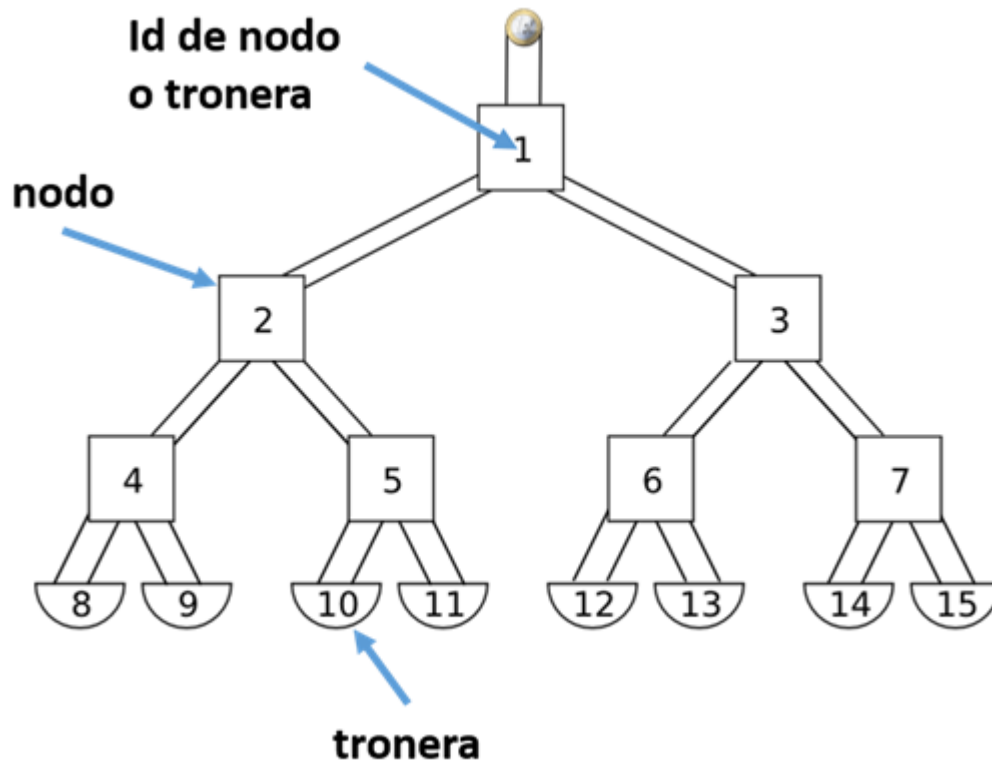
Fecha de entrega: hasta el 22/12/2016 23:59 hs.

Condiciones de entrega

1. El presente trabajo práctico deberá resolverse en grupo de hasta 3 (tres) personas.
2. Entrega: Se realizará por medio del campus virtual de la UTN, en la tarea correspondiente al TP 3. La extensión del archivo será .zip o .rar, de acuerdo al programa de compresión usado. El nombre del archivo se consigue concatenando un prefijo del número del TP con los apellidos de los integrantes separados por guiones (Ej: Pérez y Abdala, el nombre será tp3-abdala-perez.zip). Note que no hay espacios en blanco ni acentos en el nombre de archivo. Dentro del archivo de entrega, deben constar los siguientes:
 - Fuentes DrRacket: Se debe entregar un archivo denominado juego.rkt
 - Los casos de prueba se entregarán en un archivo de texto, no deben ser capturas de pantalla. Deberán cubrir diferentes resultados que puedan obtenerse de la evaluación de los predicados solicitadas. Se enfatiza que se adjunten casos de prueba que sean claros, válidos y suficientes para poder probar el trabajo. Entregar este archivo con con el nombre casos-de-prueba.txt.
 - Archivo de texto (INTEGRANTES.txt) con una línea para cada integrante en la cual figure el nombre del alumno/a y su dirección de email.
4. Penalización por entrega fuera de término: Si el trabajo práctico se entrega después de la fecha indicada, y hasta una semana tarde, tendrá una quita de 15 puntos. No serán recibidos trabajos luego de una semana de la fecha de entrega. Los trabajos que se deban rehacer/corregir fuera de la fecha de entrega tienen una quita de 30 puntos.

Descripción del problema

En el juego de la máquina receptora de fichas (o monedas), éstas ingresan a la máquina de a una por vez. La 1er. ficha que ingresa a un **nodo** es luego dirigida hacia el nodo de la izquierda, la 2da. ficha hacia el nodo de la derecha, la 3er. ficha nuevamente hacia el nodo de la izquierda, la 4ta. ficha es dirigida hacia el nodo de la derecha y así sucesivamente. Con esta lógica las fichas recorren la máquina y finalmente se acumulan en las **troner**as. En la figura que se encuentra a continuación se esquematiza la máquina descrita.



Para representar el funcionamiento de la máquina se asociará a cada nodo un estado con valor **0** (cero) o **1** (uno). Si un nodo está en estado **0** la próxima ficha que ingrese se dirigirá hacia el nodo (o tronera) sucesor de la izquierda, y si se encuentra en estado **1**, la próxima ficha que ingrese se dirigirá hacia el nodo (o tronera) sucesor de la derecha.

Este escenario es inicialmente semejante al utilizado para realizar el TP de Programación Lógica. Pero se le solicite que le incorpore una variante. Cada nodo deberá indicar explícitamente su modo de funcionamiento. Los nodos tendrán un comportamiento como el indicado en los párrafos anteriores, si están etiquetados con el símbolo '**izq**'.

Si por el contrario el nodo está etiquetado con el símbolo '**der**', y está en estado **0** la próxima ficha que ingrese se dirigirá hacia el nodo (o tronera) sucesor de la **derecha**, y si se encuentra en estado **1**, la próxima ficha que ingrese se dirigirá hacia el nodo (o tronera) sucesor de la **izquierda**.

La representación de la máquina se debe implementar como una lista de 3 elementos:

```
<maquina> := (<raiz> <maquina> <maquina>) |
```

```
(<raiz> <tronera> <tronera>)
```

```
<raiz> := ((<ld> <modo>) <estado>)
```

```
<ld> := entero >= 1
```

```
<modo> := izq | der
```

```
<estado> := 0 | 1
```

```
<trонера> ::= (trонера <Id> <numero_de_fichas>)
<numero_de_fichas> ::= entero >= 0
```

A la descripción hay que agregarle las siguientes restricciones:

<Id> ; Id es un número entero con el que se identifica el nodo o la tronera. El nodo raíz de la máquina tiene Id = 1, los siguientes nodos tienen un Id que se calcula de la siguiente manera:

$$\text{Id nodo izquierdo} = \text{Id nodo antecesor} * 2$$

$$\text{Id nodo derecho} = \text{Id nodo izquierdo} + 1.$$

Se le solicita que defina las siguientes funciones:

1. **(maquinaEstadoInicial NumeroNiveles)** ; el objetivo es que retorne una máquina con un número de niveles igual a **NumeroNiveles**, con todos los nodos en modo 'izq' y en estado 0 (cero), y las troneras con 0 (cero) fichas. Fundamentalmente se pretende que esta función se utilice para generar máquinas válidas en el estado previo a que ingrese alguna ficha.

Ejemplos:

> **(maquinaEstadoInicial 1)**

```
'(((1 izq) 0) (trонера 2 0) (trонера 3 0))
```

> **(maquinaEstadoInicial 2)**

```
'(((1 izq) 0) (((2 izq) 0) (trонера 4 0) (trонера 5 0)) (((3 izq) 0) (trонера 6 0) (trонера 7 0)))
```

> **(maquinaEstadoInicial 3)**

```
'(((1 izq) 0) (((2 izq) 0) (((4 izq) 0) (trонера 8 0) (trонера 9 0)) (((5 izq) 0) (trонера 10 0) (trонера 11 0))) (((3 izq) 0) (((6 izq) 0) (trонера 12 0) (trонера 13 0)) (((7 izq) 0) (trонера 14 0) (trонера 15 0))))
```

> **(maquinaEstadoInicial 4)**

```
'(((1 izq) 0)
  (((2 izq) 0) (((4 izq) 0) (((8 izq) 0) (trонера 16 0) (trонера 17 0)) (((9 izq) 0) (trонера 18 0) (trонера 19 0))) (((5 izq) 0) (((10 izq) 0) (trонера 20 0) (trонера 21 0)) (((11 izq) 0) (trонера 22 0) (trонера 23 0))))
  (((3 izq) 0) (((6 izq) 0) (((12 izq) 0) (trонера 24 0) (trонера 25 0)) (((13 izq) 0) (trонера 26 0) (trонера 27 0))) (((7 izq) 0) (((14 izq) 0) (trонера 28 0) (trонера 29 0)) (((15 izq) 0) (trонера 30 0) (trонера 31 0)))))
```

2. **(ingresaFicha MaquinaPrevia)** ; **MaquinaPrevia** es la máquina en el estado previo al ingreso de una ficha. La función retorna una máquina en el estado resultante de ingresar una ficha.

Ejemplos:

> **(ingresaFicha (maquinaEstadoinicial 1))**

'(((1 izq) 1) (troner 2 1) (troner 3 0))

> **(ingresaFicha (maquinaEstadoinicial 2))**

'(((1 izq) 1) (((2 izq) 1) (troner 4 1) (troner 5 0)) (((3 izq) 0) (troner 6 0) (troner 7 0)))

> **(ingresaFicha (maquinaEstadoinicial 3))**

'(((1 izq) 1) (((2 izq) 1) (((4 izq) 1) (troner 8 1) (troner 9 0)) (((5 izq) 0) (troner 10 0) (troner 11 0))) (((3 izq) 0) (((6 izq) 0) (troner 12 0) (troner 13 0)) (((7 izq) 0) (troner 14 0) (troner 15 0))))

> **(ingresaFicha (ingresaFicha (maquinaEstadoinicial 2)))**

'(((1 izq) 0) (((2 izq) 1) (troner 4 1) (troner 5 0)) (((3 izq) 1) (troner 6 1) (troner 7 0)))

3. **(ingresaVariasFichas MaquinaPrevia NFichas)** ; retorna como resultado la máquina resultante del ingreso de **NFichas** fichas en **MaquinaPrevia**.

Ejemplos:

> **(ingresaVariasFichas (maquinaEstadoinicial 2) 3)**

'(((1 izq) 1) (((2 izq) 0) (troner 4 1) (troner 5 1)) (((3 izq) 1) (troner 6 1) (troner 7 0)))

> **(ingresaVariasFichas (maquinaEstadoinicial 2) 7)**

'(((1 izq) 1) (((2 izq) 0) (troner 4 2) (troner 5 2)) (((3 izq) 1) (troner 6 2) (troner 7 1)))

> **(ingresaVariasFichas (maquinaEstadoinicial 3) 10)**

'(((1 izq) 0) (((2 izq) 1) (((4 izq) 1) (troner 8 2) (troner 9 1)) (((5 izq) 0) (troner 10 1) (troner 11 1))) (((3 izq) 1) (((6 izq) 1) (troner 12 2) (troner 13 1)) (((7 izq) 0) (troner 14 1) (troner 15 1))))

4. **(resultado MaquinaPrevia NFichas)** ; retorna una lista con las troneras resultantes de ingresar un número de fichas igual a **NFichas** en **MaquinaPrevia**.

Ejemplos:

> **(resultado (maquinaEstadoinicial 2) 3)**

'((troner 4 1) (troner 5 1) (troner 6 1) (troner 7 0))

> **(resultado (maquinaEstadoinicial 2) 7)**

'((troner 4 2) (troner 5 2) (troner 6 2) (troner 7 1))

5. **(cambiarModoNodo MaquinaPrevia IdNodo Modo)** ; retorna la máquina resultante de cambiar el modo de comportamiento del nodo **IdNodo**. El **Modo** puede ser 'ízk o 'der.

> **(define M (maquinaEstadoInicial 2))**

> **M**

'(((1 ízk) 0) (((2 ízk) 0) (trónera 4 0) (trónera 5 0)) (((3 ízk) 0) (trónera 6 0) (trónera 7 0)))

> **(define M1 (cambiarModoNodo M 3 'der))**

> **M1**

'(((1 ízk) 0) (((2 ízk) 0) (trónera 4 0) (trónera 5 0)) (((3 der) 0) (trónera 6 0) (trónera 7 0)))

> **(ingresaVaríasFichas M 2)**

'(((1 ízk) 0) (((2 ízk) 1) (trónera 4 1) (trónera 5 0)) (((3 ízk) 1) (trónera 6 1) (trónera 7 0)))

> **(ingresaVaríasFichas M1 2)**

'(((1 ízk) 0) (((2 ízk) 1) (trónera 4 1) (trónera 5 0)) (((3 der) 1) (trónera 6 0) (trónera 7 1)))