

# Clase Auxiliar N°2 - CC30A

12 de abril de 2007

## 1. El Problema de Josephus.

Dado un grupo de  $n$  hombres ordenados en círculo. Un verdugo irá matando cada  $m$ -ésimo hombre alrededor del círculo hasta que quede sólo un sobreviviente. Encuentre la posición en que debería colocarse el último sobreviviente.

Haga un método que vaya eliminando los  $m$ -ésimos nodos de una lista enlazada, hasta que quede sólo uno. Luego, retorne la posición inicial del último nodo sobreviviente.

Considere la siguiente representación de un nodo:

```
class Nodo{
int posicion;
Nodo sgte;
public Nodo( int posicion , nodo sgte ) { ... }
}
```

## 2. UNDO y REDO

(Pregunta 4 Control 1 2006) Suponga que un sistema debe procesar una secuencia de acciones, a cada una de las cuales podría eventualmente hacersele “undo”. Diseñe un esquema que permita ir procesando las acciones en el orden en que vienen llegando, y que sea capaz también de ejecutar instrucciones UNDO (para deshacer la última acción procesada) y REDO (para re-ejecutar la última a la que se le hizo UNDO). Se le sugiere utilizar dos stacks (pilas): uno para las acciones que podrán ser objeto de UNDO y otro para las que podrían ser objeto de un REDO. Indique claramente (en palabras y con diagramas) cómo se modifican estos stacks:

1. al procesarse una acción dada,
2. al ejecutarse un UNDO, y
3. al ejecutarse un REDO

Indique en cada caso las condiciones para que sea válido ejecutar esas acciones.

### **3. Notación polaca a árbol binario.**

Implemente un método que reciba como parámetro un string que describe a un árbol binario en notación polaca (postorden) y lo almacene en un árbol binario. Los nodos externos están representados con puntos y los nodos internos con strings de largo 1.