

Orientación a Objetos I - Curso 2011 - Parcial
Sábado 26 de Noviembre

Se desea modelar redes de sensores inalámbricos. Las mismas están compuestas por dos tipos diferentes de elementos: **Sensores y Hubs**.

Cada componente de la red conoce a sus vecinos (la relación es simétrica) a los que les puede mandar mensajes. Los sensores guardan un estado (encendido / apagado), mientras que los Hubs no, ya que sólo retransmiten mensajes a sus vecinos. Además, los Sensores y los Hubs poseen una fecha de fabricación y también conocen una posición que se expresa con una instancia de la clase Point de Smalltalk.

Se desea modelar e implementar la funcionalidad de Encendido / Apagado de toda la red y ciertas consultas sobre la misma. Tanto los sensores como los hubs deben responder al siguiente protocolo:

#switchOn

"Si el sensor está apagado, se enciende y propaga el mensaje a los vecinos.

Un Hub siempre propaga el mensaje de encendido"

#switchOff

"Si el sensor está encendido, se apaga y propaga el mensaje a los vecinos.

Un Hub siempre propaga el mensaje de apagado"

#fechaDefabricaciónDelElementoMasViejo

"Retorna la fecha mas antigua de entre todos los sensores y hubs"

#posicionesDeElementos

"Retorna una colección de puntos, que son las posiciones de todos los elementos (sensores / hubs) de la red."

Considere que: (i) la red es una componente conexa, es decir, no hay sensores / hubs que no se conectan con la red. (ii) La posición de cada sensor / Hub es única.

Considere el siguiente template para declarar sus clases y métodos.

```
<Superclass> subclass: <NombreDeMiClase>  
instanceVariables: '<variable1> <variable2>'
```

```
MyClass>>esteEsUnMetodoDeInstancia  
"implementación del método"
```

```
MyClass class>> esteEsUnMetodoDeClase  
"implementación del método"
```

Tareas

1. Realice el diagrama de clases UML
2. Realice el diagrama de secuencia en donde se le envía el mensaje #switchOn al sensor1 de la red representada por sensor1<->hub1<->sensor2. Considere que ambos sensores están apagados.
3. Implemente en Smalltalk las clases y los métodos necesarios para cumplir con el protocolo definido, como así también para instanciar redes.
4. Escriba el código Smalltalk necesario para instanciar la red del punto 2 en un workspace.