

Examen de Prácticas

Asignatura: “Prácticas de Programación”

Septiembre 2006

Se desea hacer un sistema para controlar una planta compuesta por un conjunto de paneles solares que producen electricidad. Cada panel solar individual se representa en el computador por medio de un objeto de la clase PanelSolar, cuya interfaz es:

```
public class PanelSolar
{
    /**
     * Constructor al que se le pasa el identificador del panel
     */
    public PanelSolar(String id) {...}

    /**
     * Retorna el identificador del panel
     */
    public String id() {...}

    /**
     * Método para alinear el panel; retorna true si se ha conseguido
     * alinear con el sol, y false en caso contrario
     */
    public boolean panelAlineado() {...}

    /**
     * Retorna el acimut del panel, en grados, que nos indica
     * hacia dónde apunta en la dirección horizontal.
     * El Sur se representa con el valor 180 grados.
     * Lanza Averiado si el panel está averiado
     */
    public double acimut() throws Averiado {...}

    /**
     * Retorna la elevación del panel, en grados, sobre el horizonte.
     * Cuando el panel apunta al horizonte el valor es 0 grados
     * Lanza averiado si el panel está averiado
     */
    public double elevacion() throws Averiado {...}

    /**
     * Retorna la potencia eléctrica, en vatios, que el panel
     * está produciendo en este instante
     */
    public double potencia() {...}
}
```

Por otro lado se dispone de la clase PlantaSolar, que responde a la siguiente interfaz:

```
public class PlantaSolar
{
    /**
     * Añade un nuevo panel solar, que se pasa como parámetro
     * Si ya existe uno con el mismo identificador lanza YaExiste
     * En otro caso, añade el panel en la lista de paneles
     */
    public void anadePanel(PanelSolar p) throws YaExiste {...}

    /**
     * Alinea todos los paneles de la planta y con los que
     * resulten correctamente alineados y no estén averiados
     * calcula el acimut y elevacion medios, almacenándolos
     * en los respectivos atributos
     */
}
```

```

*/
public void alinea() {...}

/**
 * Encuentra y retorna un panel anómalo, que es aquel que sin
 * estar averiado, su acimut o elevación
 * se diferencia en más de un 5% de los valores medios
 * almacenados en los atributos
 * Retorna null si no hay paneles que cumplan esa condición
 */
public PanelSolar panelAnomalo() {... }
}

```

Las excepciones están definidas en clases aparte de la forma:

```

public class Averiado extends Exception{}
public class YaExiste extends Exception{}

```

Deseamos hacer la prueba de la clase `PlantaSolar` para comprobar su correcto funcionamiento, en condiciones simuladas. Para ello, vamos a implementar la clase `PanelSolar` de modo que opere de forma simulada. Sus métodos deben hacer lo siguiente:

- *Constructor*: además de almacenar el identificador en un atributo, el constructor debe pedir al usuario que introduzca por pantalla el acimut elevación y potencia que devolverán respectivamente los métodos `acimut()`, `elevacion()` y `potencia()`, el valor que debe devolver el método `panelAlineado()`, y el hecho de que el panel esté averiado o no. Estos datos se guardarán en atributos creados al efecto.
- `id()`: retorna el identificador almacenado en un atributo
- `panelAlineado()`: retorna el valor del atributo que contiene el valor (leído del teclado) que debe devolver este método
- `acimut()`: si el panel está averiado, lanza `Averiado`; en caso contrario, retorna el valor de elevación almacenado en un atributo
- `elevacion()`: si el panel está averiado, lanza `Averiado`; en caso contrario, retorna el valor de elevación almacenado en un atributo
- `potencia()`: retorna la potencia almacenada en un atributo

Por otro lado, debe hacerse un programa principal de prueba que haga lo siguiente

- Crear cuatro paneles solares
- Crear una planta solar y añadirle los cuatro paneles solares
- Intentar añadir de nuevo uno de los paneles anteriores y tratar la excepción `YaExiste` poniendo un mensaje en pantalla
- Alinear la planta solar
- Mostrar en pantalla los valores de acimut y elevación media
- Mostrar en pantalla el identificador del primer panel anómalo, si lo hay, o un mensaje diciendo que no hay paneles anómalos, si no los hay.

El profesor te entregará el software que está ya realizado.

Nota: Cada parte del ejercicio se valorará en función de su dificultad:

- clase `PanelSolar`: 40% de la nota
- programa de prueba: 60% de la nota