

PRIMERA PRUEBA DE EVALUACIÓN

Estos ejercicios resueltos deben enviárselo al profesor tutor que tenga asignado en su grupo de tutoría, y si no tiene profesor tutor a los profesores de la sede central.

Se recomienda que se realicen en la semana que se proponen. En todo caso para poderse considerar en la calificación final debe enviarse antes del 15 de Junio. Si tiene problemas para enviarlos en fecha contacte con su profesor tutor.

Use el foro de su grupo de tutoría para comentarios y cuestiones sobre los ejercicios de evaluación continua.

Ejercicio 1

Para la programación de las N-versiones (N-versión programming), contestar a las siguientes preguntas:

- a) Explicar en qué suposición se basa, cuando deja de ser válida y que hay que realizar para que la suposición sea cierta.
- b) ¿De qué es responsable el proceso director (driver process)?
- c) ¿Qué se entiende por votación inexacta (inexact voting)? Como se puede resolver y en que consiste el problema de la comparación consistente (consistent comparison problem).
- d) Enumerar y comentar los tres factores principales del éxito de la programación de las N-versiones.

Ejercicio 2

En una aplicación de control, se calienta gas en una cámara cerrada. La cámara se encuentra rodeada por un refrigerante que reduce la temperatura del gas por conducción. También hay una válvula que, al ser abierta, libera el gas en la atmósfera. La operación del proceso está controlada por un paquete Java cuya especificación se proporciona a continuación. Por razones de seguridad, el paquete reconoce diversas condiciones de error; éstas se notifican al usuario mediante la generación de excepciones. Cuando se es incapaz de apagar el calentador se genera la excepción `HeaterStuckOn` desde el método `heaterOff`. Cuando se es incapaz de bajar la temperatura del gas incrementando el caudal de refrigerante se genera la excepción `TemperatureStillRising` desde el método `increaseCoolant`. Finalmente, si se es incapaz de liberar el gas hacia la atmósfera se genera la excepción `ValveStuck` desde el método `openValve`.

```
public class ValveStuck extends Exception;

public class HeaterStuckOn extends Exception;
```

```
public class TemperatureStillRising extends Exception;

public class TemperatureControl
{
    public void heaterOn ();
    public void heaterOff () throws HeaterStuckOn;
    public void increaseCoolant ()
        throws TemperatureStillRising ;
    public void openValve () throws ValveStuck;
    public void panic ();
}
```

Escriba un método que al ser llamado intente apagar el calentador de la cámara de gas. Si el calentador está atascado y encendido aumentará el caudal de refrigerante que rodea la cámara. Si la temperatura continúa aumentando, entonces deberá abrir la válvula de escape para soltar el gas. Si esto falla, deberá hacer sonar una alarma e informar a los servicios de emergencia.