

Programación Concurrente y de Tiempo Real
Grado en Ingeniería Informática
Examen Final de la Asignatura
Febrero de 2012

Apellidos:

Nombre:

D.N.I.:

Grupo:

1 Notas

1. Escriba su nombre, apellidos, d.n.i. y grupo con letra clara y legible en el espacio habilitado para ello, y en todos los folios blancos que utilice.
2. Firme el documento en la esquina superior derecha de la primera página.
3. Dispone de diez minutos para leer los enunciados y formular preguntas o aclaraciones sobre ellos. Transcurrido ese tiempo, no se contestarán preguntas. Dispone de 2:00 horas para completar el ejercicio.
4. No complete el documento a lápiz. Utilice bolígrafo o rotulador.
5. Utilice los folios blancos que se le proporcionan para resolver los enunciados, pero traslade a este documento únicamente la solución final que obtenga, utilizando el espacio específicamente habilitado para ello, sin sobrepasarlo en ningún caso, y sin proporcionar información o respuestas no pedidas. Entregue tanto el enunciado como los folios blancos.

2 Criterios de Corrección

1. El examen se calificará de cero a diez puntos, y ponderará en la calificación final al 90% bajo los supuestos recogidos en la ficha de la asignatura.
2. Cada enunciado incluye información de la puntuación que su resolución correcta representa, incluida entre corchetes.
3. Un enunciado (cuestión teórica o problema) se considera correcto si la solución dada es correcta completamente. En cualquier otro caso se considera incorrecto y no puntúa.

4. Un enunciado de múltiples apartados (cuestión teórica o problema) es correcto si y solo si todos los apartados que lo forman se contestan correctamente. En cualquier otro caso se considera incorrecto y no puntúa

3 Cuestiones Teóricas de Opción Simple

Conteste VERDADERO o FALSO a las siguientes aserciones. Marque con un aspa (así ☒) la respuesta que proceda. Si se equivoca, tache y vuelva a marcar nuevamente. Cada cuestión correcta puntúa 0.1 puntos. **Se muestra un ejemplo de este tipo de cuestión.**

1. Es posible registrar en un DNS rmiregistry objetos servidores que ofrezcan diferentes servicios desde máquinas virtuales diferentes. ☐ V ☐ F
2. ...

4 Cuestiones Teóricas de Opción Múltiple

Escoja la respuesta correcta rodeándola con un círculo. Si se equivoca, tache y vuelva a marcar nuevamente. Únicamente una de las respuestas es correcta. Cada cuestión correcta puntúa 0.2 puntos. **Se muestra un ejemplo de este tipo de cuestión.**

1. Determine en cuál de las siguientes situaciones existe concurrencia
 - a) Dos procesos que se ejecutan de forma simultánea.
 - b) Dos hilos que se ejecutan de forma simultánea. (De la vuelta a la hoja ahora)
 - c) Dos procesos que activan dos hilos concurrentes y que se ejecutan de forma simultánea.
 - d) En todos los casos anteriores.
 - e) Ninguna de las respuestas anteriores es correcta.
2. ...

5 Cuestiones de Desarrollo Corto

Conteste a las preguntas que le formulan en el espacio habilitado para ello. Deberá razonar o justificar su respuesta siempre que se le indique. La ausencia del razonamiento o de la justificación invalidarán la respuesta al no ser esta completa. Cada cuestión correcta puntúa 0.2 puntos. **Se muestran algunos ejemplos de este tipo de cuestión**

1. Dado el siguiente programa escrito en java, indique qué salida proporcionará. Razone su respuesta.

```
public class incConcurrente
    extends Thread
{
    private static int n = 0;
    private int nIter;
    public incConcurrente(int v) {nIter=v;}
    public void run()
    {
```

```

        for(int i=1;i<=nIter;i++) n++;
    }
    public static void main(String[] args)
    throws InterruptedException
    {
        int p = 10;
        incConcurrente[] h = new incConcurrente[p];
        for(int i=0;i<h.length;i++)h[i] = new incConcurrente(1000);
        for(int i=0;i<h.length;i++)h[i].start();
        for(int i=0;i<h.length;i++)h[i].join();
        System.out.print(n);
    }
}

```

Escriba aquí su respuesta:

2. Dado el siguiente programa escrito en java, indique qué salida produce. Justifique su respuesta.

```

class Animal{
    public void move(){
        System.out.println("Los animales se mueven");
    }
}
class Perro extends Animal{
    public void move(){
        System.out.println("Los perros andan y corren");
    }
}
public class PruebaPerro{
    public static void main(String args[]){
        Animal a = new Animal();
        Animal b = new Perro();
        a.move();
        b.move();
    }
}

```

Escriba aquí su respuesta:

3. Dado el siguiente programa escrito en java, indique qué salida produce. Razone su respuesta.

```
import java.util.concurrent.locks.*;
public class examen implements Runnable
{
    public static Integer p = new Integer(0);
    public static ReentrantLock r = new ReentrantLock();
    public int vMax;
    public examen(int v) {vMax = v;}
    public void run()
    {
        Object q=p;
        for(int cont=0;cont<vMax; cont++)
        { r.lock(); p++; r.unlock();
          synchronized(q){try{q.wait();}
            catch(InterruptedException e){}}
        }
    }
    public static void main(String[] args)
        throws InterruptedException
    {
        Thread [] h = new Thread[10];
        for(int i=0;i<h.length;i++)h[i]=new Thread(new examen(100));
        for(int i=0;i<h.length;i++)h[i].start();
        Thread j = Thread.currentThread();
        j.sleep(1000);
        p++;
        System.out.print(p);
    }
}
```

Escriba aquí su respuesta:

4. ...

6 Problemas

Desarrolle a continuación los enunciados que se le plantean.

1. Se desea disponer de un protocolo que permita transformar un monitor teórico tipo *Hoare* a una clase java con igual comportamiento, utilizando cerrojos de clase `ReentrantLock`. [2.6 puntos]

Se pide:

- a) Desarrollar el protocolo.
- b) Escribir un monitor *Hoare* con un único procedimiento de nombre `input()` tal que los tres primeros procesos externos que lo llamen queden suspendidos y el cuarto los despierte, repitiéndose este comportamiento de forma cíclica.
- c) Aplicar el protocolo del primer apartado al monitor obtenido en el segundo para obtener un monitor java equivalente.

2. Un estudio de arquitectura dispone de un sistema con n puestos de trabajo que tienen acceso simultáneo a k plotters. En cada puesto de trabajo se ejecuta un software de CAD que requerirá el uso de alguno de los plotters. Desarrolle utilizando semáforos *Dijkstra* el código de dicho proceso software de manera que se cumplan las dos siguiente condiciones: [2.6 puntos]

- el acceso a cada impresora individual tiene lugar en exclusión mutua.
- puede haber varios procesos imprimiendo en varias impresoras, siempre que sea en distintas impresoras.

(Utilice este espacio discrecionalmente para escribir su soluciones a los problemas)