

PROGRAMACIÓN AVANZADA

Convocatoria extraordinaria – 1 de julio de 2016

Normas de Examen:

No se recogerá el enunciado del examen.

En cada pregunta se indica el valor de la misma sobre la nota final.

La duración del examen es de **dos horas y media**

Se numerarán todas las hojas con el número de hoja y el total (ejemplo: 1/4, 2/4, 3/4, 4/4).

1) (2 puntos) Elegir la respuesta correcta (cada pregunta correcta puntúa 0,2, las no respondidas no puntúan y las erróneas puntúan -0,2).

1. Los niveles de cumplimiento de un paradigma hacen referencia...
 - a) a que los lenguajes no siempre cumplen perfectamente las especificaciones del paradigma
 - b) a la calidad de un lenguaje de programación y sus estructuras
 - c) al compromiso con la normativa general de especificaciones de abstracción
 - d) a que los paradigmas no siempre cumplen las especificaciones sintácticas existentes
2. Los paradigmas estructurado y orientado a objetos, se caracterizan por...
 - a) proponer un código de mayor nivel de abstracción que la programación imperativa
 - b) previenen a un lenguaje de la utilización del paradigma declarativo
 - c) son paradigmas de estructura que nos proponen cómo organizar nuestro código
3. El uso más común de los paradigmas es el de...
 - a) describir los entornos de desarrollo y sus herramientas
 - b) comparar la eficiencia de los lenguajes y de sus implementaciones
 - c) optimizar compiladores, intérpretes y entornos de desarrollo
 - d) describir lenguajes de forma simplificada y eficaz en función de los paradigmas que cumplen
4. El uso de concurrencia...
 - a) no abarata costes pero acelera el desarrollo
 - b) abarata costes pero complica el modelado de los programas
 - c) abarata costes y simplifica el modelado de programas
 - d) capacita para atender varias solicitudes pero complica el modelado de programas
5. La seguridad/fiabilidad de los programas...
 - a) se pone en riesgo por el acceso simultáneo a elementos compartidos
 - b) aumenta con la utilización de la concurrencia porque los datos están menos tiempo en uso
 - c) se complica por el uso de exclusión mutua
6. Para la clasificación de Flynn, los ordenadores se caracterizan por...
 - a) el uso del pipelining y su número
 - b) el número de bits procesados simultáneamente
 - c) el tipo de bus de datos de acceso a la memoria
 - d) el número de CUs y ALUs
7. La exclusión mutua debe aplicarse:
 - a) siempre que se utilicen hilos o procesos
 - b) cuando existan secciones críticas sobre la misma variable
 - c) para evitar que la sincronización de los hilos más rápidos produzca inanición en los más lentos
 - d) en los modelos de comunicación pura
8. La comunicación sincronizada...
 - a) intercambia información y lo hace con coordinación en el tiempo
 - b) coordina los procesos para evitar que se ejecuten simultáneamente
 - c) asegura la exclusión mutua de procesos asociados a las mismas variables
 - d) integra los procesos que contienen las mismas secciones críticas
9. La seguridad en sistemas distribuidos
 - a) Disminuye y es una desventaja que limita la propia existencia de los sistemas
 - b) Aumenta y es una ventaja que hace imprescindible el uso de distribución
 - c) Aumenta en función del número de nodos conectados
 - d) Disminuye por las características de los sistemas pero puede aumentar si se planifica bien
10. Las llamadas a procedimientos remotos o RPC:
 - a) Son un sistema obsoleto de comunicar sistemas
 - b) Establece un nivel de abstracción mayor en la comunicación entre sistemas
 - c) Dejan de ser posibles una vez establecidos los sistemas de enrutamiento modernos
 - d) Proporcionan un mayor control de los detalles técnicos de la comunicación

- 2) (1 punto) Definir qué es una situación de interbloqueo y las 4 condiciones necesarias para que se produzca. Indicar posibles consecuencias de esta situación y posibles soluciones (utilizar como máximo una cara).
- 3) (0,5 puntos) Indicar y explicar qué saldría por pantalla al llamar al siguiente método (utilizar como máximo una cara):

```
...
public void metodo () {
    while (true) {
        try {
            wait ();
            Thread.currentThread().interrupt ();
        } catch (InterruptedException e) {
            System.out.println("Hilo interrumpido");
        }
        i++;
    } //Fin while
} //Fin metodo
...
```

- 4) (0,5 puntos) Explicar qué son las actividades en Android, para qué sirven y qué contienen (utilizar como máximo una cara).
- 5) (3 puntos) Se desea modelar el comportamiento de un grupo de jugadores de fútbol que tienen un partido concertado y los árbitros que, además, son los responsables de las instalaciones. Se cuenta con 22 jugadores que van llegando de forma escalonada (tiempo aleatorio entre 5-20s) y los 5 árbitros que, desde el principio están en las instalaciones, esperando a que todos lleguen. Una vez reunidos los 22 jugadores deben esperar a que los 5 árbitros se preparen (tardan de 2s a 7s).

A continuación comienza el partido, donde los 22 jugadores se ponen en marcha (transcurren 90s).

Los árbitros esperan a que todos los jugadores terminen y los 5 árbitros finalizan con un mensaje indicando el final del partido.

Se mostrará en pantalla cada cambio de estado, incluyendo la finalización de cada estado y el comienzo del siguiente.

Se pide: desarrollar el código completo del programa. El único elemento static del programa será el método main. El programa deberá funcionar a excepción de posibles errores menores "de compilación".

- 6) (3 puntos) Crear tres programas Java que integra 3 equipos informáticos (E1, E2 y E3) en un sistema distribuido con la siguiente funcionalidad: la comunicación E1->E2 deberá realizarse mediante canales de comunicación asíncronos, la comunicación entre E2->E3 será con RMI, mientras que la comunicación E1<-E3 será mediante sockets orientados a conexión.

El equipo E1, contiene un programa que interactúa con el usuario, le pide datos numéricos enteros positivos y envía cada uno tras ser leído al equipo E2. Se le piden un total de 5 números al usuario, pero si introduce alguno negativo, no se cuenta ni se procesa y se pide otro (al final se tienen que haber leído y enviado 5 enteros positivos). Cuando se termina el proceso de leer y enviar al proceso E2, se queda a la espera de recibir del proceso E3 una serie de números indefinida que acaba cuando se recibe un número negativo. Los números recibidos se imprimen por pantalla salvo el negativo y el programa E1 finaliza su ejecución.

El equipo E2 deberá implementarse como servicio (nunca finalizará su ejecución) recibiendo datos numéricos (desde el equipo E1). Para cada número recibido se generan 2 números, multiplicando por 2 y dividiendo por 2 el original (división entera, sin decimales). Dado que se recibirán 5 números, se generan 10 y se enviarán al equipo E3. Se realiza el envío de forma inmediata tras la generación de cada par de números.

El equipo E3 recibirá 10 datos numéricos y realizará un filtro eliminando los números que sean múltiplos de 3 y enviando los restantes al equipo E1 seguidos de un -10 para indicar el fin de la ejecución de E1. El envío se realizará al final del proceso de recepción, cuando los 10 datos hayan sido recibidos.

Se pide: desarrollar el código completo del programa. El único elemento static del programa será el método main. El programa deberá funcionar a excepción de posibles errores menores "de compilación".