

PROGRAMACION CONCURRENTE - EXAMEN FINAL 3-3-2010

En todos los casos, responda con claridad y sintéticamente.

En los casos que corresponda, NO SE CONSIDERARAN RESPUESTAS SIN JUSTIFICAR.

Es condición necesaria para la aprobación que el ejercicio 10 esté bien resuelto.

Tiempo Máximo 2 hs 15 min.

1. a) Defina programa concurrente, programa paralelo y programa distribuido.
- b) Compare la comunicación por mensajes sincrónicos y asincrónicos en cuanto al grado de concurrencia y la posibilidad de entrar en deadlock.
- c) Defina el concepto de “continuidad conversacional” entre procesos

2. Cómo pueden clasificarse las arquitecturas multiprocesador:

- a) según el mecanismo de control? Describa.
- b) según la organización del espacio de direcciones? Describa.
- c) según la red de interconexión? Describa.

3. Suponga los siguientes programas concurrentes. Asuma que “funcion” existe, y que los procesos son iniciados desde el programa principal.

P1	<pre>chan canal (double) process grano1 { int veces, i; double sum; for [veces= 1 to 10] { for [i = 1 to 10000] sum=sum+funcion(i); send canal (sum); } }</pre>	<pre>process grano2 { int veces; double sum; for [veces= 1 to 10] { receive canal (sum); printf (sum); } }</pre>	P2	<pre>chan canal (double) process grano1 { int veces, i; double sum; for [veces= 1 to 10000] { for [i = 1 to 10] sum=sum+i; send canal (sum); } }</pre>	<pre>process grano2 { int veces; double sum; for [veces= 1 to 10000] { receive canal (sum); printf (sum); } }</pre>
----	---	--	----	--	---

- a) Analice desde el punto de vista del número de mensajes.
- b) Analice desde el punto de vista de la granularidad de los procesos.
- c) Cuál de los programas le parece más adecuado para ejecutar sobre una arquitectura de tipo cluster de PCs? Justifique.

4. Indiqué por qué puede considerarse que existe una dualidad entre los mecanismos de monitores y pasaje de mensajes. Ejemplifique.

5. Dado el siguiente programa concurrente con memoria compartida:

```
x = 3; y = 2; z = 5;
co
    x = y * z // z = z * 2    // y = y + 2x
oc
```

- a) Cuáles de las asignaciones dentro de la sentencia co cumplen la propiedad de “A lo sumo una vez”. Justifique claramente.
- b) Indique los resultados posibles de la ejecución. Justifique.
- Nota 1: las instrucciones NO SON atómicas.
- Nota 2: no es necesario que liste TODOS los resultados, pero sí todos los casos que resulten significativos.

6. a) Defina el concepto de “sincronización barrier”.Cuál es su utilidad?
- b) Qué es una barrera simétrica?
- c) Describa “combining tree barrier” y “butterfly barrier”. Marque ventajas y desventajas en cada caso.
7. a) Describa brevemente en qué consisten los mecanismos de RPC y Rendezvous. Para qué tipo de problemas son más adecuados?
- b) Por qué es necesario proveer sincronización dentro de los módulos en RPC? Cómo puede realizarse esta sincronización?
- c) Qué elementos de la forma general de rendezvous no se encuentran en el lenguaje ADA?
8. a) Cuál es el objetivo de la programación paralela?
- b) Defina las métricas de speedup y eficiencia. Cuál es el significado de cada una de ellas (qué miden)? Ejemplifique.
- c) Suponga que la solución a un problema es paralelizada sobre p procesadores de dos maneras diferentes. En un caso, la eficiencia está regido por la función $E=1/p$ y en el otro por la función $E= 1/p^2$. Cuál de las dos soluciones se comportará más eficientemente al crecer la cantidad de procesadores? Justifique.
- d) Suponga que el tiempo de ejecución de un algoritmo secuencial es de 10000 unidades de tiempo, de las cuales sólo el 80% corresponde a código paralelizable. Cuál es el límite en mejora que puede obtenerse paralelizando el algoritmo?
9. Explique sintéticamente los 7 paradigmas de interacción entre procesos en programación distribuida. En cada caso ejemplifique, indique qué tipo de comunicación por mensajes es más conveniente y cuál es la arquitectura de hardware que se ajusta mejor? Justifique sus respuestas.
10. Suponga n^2 procesos organizados en forma de grilla cuadrada. Cada proceso puede comunicarse solo con los vecinos izquierdo, derecho, de arriba y de abajo (los procesos de las esquinas tienen solo 2 vecinos, y los otros en los bordes de la grilla tienen 3 vecinos). Cada proceso tiene inicialmente un valor local v .
- a) Escriba un algoritmo heartbeat que calcule el máximo y el mínimo de los n^2 valores. Al terminar el programa, cada proceso debe conocer ambos valores.
- b) Analice la solución desde el punto de vista del número de mensajes.
- c) Puede realizar alguna mejora para reducir el número de mensajes?