

## PROGRAMACION CONCURRENTE - EXAMEN FINAL - -2009

En todos los casos, responda con claridad y sintéticamente.

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

Tiempo Máximo 2 hs 15 min.

1. Dado el siguiente programa concurrente con memoria compartida:

```
x = 2; y = 3; z = 4;
co
  x = x - 2 // y = z * x // z = z + y
oc
```

a) En qué consiste la propiedad de “A lo sumo una vez” (ASV)?

b) Cuáles de las asignaciones dentro de la sentencia co cumplen la propiedad de ASV. Justifique claramente.

c) 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.

2.

a) A qué se denomina propiedad de programa? Qué son las propiedades de vida y seguridad? Ejemplifique.

b) Defina fairness. Relacione dicho concepto con las políticas de scheduling.

c) Cuáles son las propiedades que debe cumplir un protocolo de E/S a una sección crítica? Cuáles son de seguridad y cuáles de vida?

d) Cuáles son los defectos que presenta la sincronización por busy waiting? Diferencie esta situación respecto de los semáforos.

e) Explique la semántica de la instrucción de grano grueso AWAIT y su relación con instrucciones tipo Test & Set o Fetch &Add.

3. Una manera de ordenar  $n$  enteros es usar el algoritmo *odd/even exchange sort* (también llamado *odd/even transposition sort*). Asuma que hay  $n$  procesos  $P[1..n]$  y que  $n$  es par. En este algoritmo cada proceso ejecuta una serie de rondas. En las rondas impares, los procesos con número impar  $P[\text{impar}]$  intercambian valores con  $P[\text{impar}+1]$  si los valores están desordenados. En las rondas con número par, los procesos con número par  $P[\text{par}]$  intercambian valores con  $P[\text{par}+1]$  si los valores están desordenados (en las rondas pares  $P[1]$  y  $P[n]$  no hacen nada).

a) Determine cuántas rondas deben ejecutarse en el peor caso para ordenar los  $n$  números.

b) Escriba (utilizando algún mecanismo de memoria compartida) un algoritmo data parallel para ordenar un arreglo de enteros  $a[1:n]$  en forma ascendente

c) Modifique el algoritmo para terminar tan pronto como el arreglo fue ordenado

d) Modifique la respuesta de a) para usar  $k$  procesos, asuma que  $n$  es múltiplo de  $k$

4.

a) Describa la técnica de “passing the baton”?Cuál es su utilidad?

b) Qué relación encuentra con la técnica “passing the condition”?

5. Describa los mecanismos de comunicación y sincronización provistos por MPI, Ada, Java y Linda.

6. Sea el problema en cual N procesos poseen inicialmente cada uno un valor V, y el objetivo es que todos conozcan cuál es el máximo y cuál es el mínimo de todos los valores.
- a) Plantee conceptualmente posibles soluciones con las siguientes arquitecturas de red: centralizada, simétrica (o totalmente conectada) y anillo circular (NO IMPLEMENTE).
  - b) Analice las soluciones desde el punto de vista del número de mensajes y la performance global del sistema.

7. Sea la siguiente solución al problema del producto de matrices de  $n \times n$  con P procesos trabajando en paralelo.

```
process worker[w = 1 to P] { # strips en paralelo (p strips de n/P filas) }
    int first = (w-1) * n/P; # Primera fila del strip
    int last = first + n/P - 1; # Ultima fila del strip
    for [i = first to last] {
        for [j = 0 to n-1] {
            c[i,j] = 0.0;
            for [k = 0 to n-1]
                c[i,j] = c[i,j] + a[i,k]*b[k,j];
        }
    }
}
```

- a) Suponga que  $n=128$  y cada procesador es capaz de ejecutar un proceso. Cuántas asignaciones, sumas y productos se hacen secuencialmente (caso en que  $P=1$ )? Cuántas se realizan en cada procesador en la solución paralela con  $P=8$ ?
- b) Si los procesadores P1 a P7 son iguales, y sus tiempos de asignación son 1, de suma 2 y de producto 3, y si P8 es 4 veces más lento, Cuánto tarda el proceso total concurrente?Cuál es el valor del speedup (Tiempo secuencial/Tiempo paralelo)?. Modifique el código para lograr un mejor speedup.