

SISTEMAS OPERATIVOS
Segundo Cuatrimestre de 2022

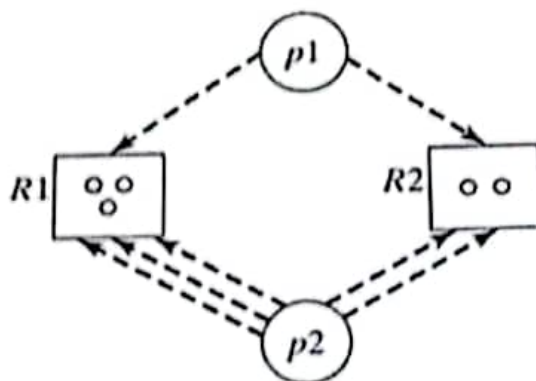
Examen Recuperatorio

Registro. [REDACTED]	Nombre [REDACTED]	Cant. H. 4
----------------------	-------------------	------------

Indicaciones

- Coloque el nombre, número de registro y cantidad de hojas entregadas sin incluir el enunciado.
- Realice UN EJERCICIO por hoja. Identifique el ejercicio en la hoja.
- No utilice lápiz. Firme la última hoja

- a) ¿Cuál es la mayor ventaja de implementar hilos en el espacio de usuario? ¿Cuál es su mayor desventaja?
- b) ¿Puede un hilo siempre ser apropiado por una interrupción de reloj? ¿Si es así, bajo que circunstancias? ¿Sino, porqué?
- a) Brinde tres ejemplo de interbloqueos que no estén vinculados al entorno de sistemas de computación.
- b) Considere el siguiente gráfico de utilización máxima de recursos. Los círculos indican recursos disponibles y las flechas necesidades. Ambos procesos comienzan sin recursos asignados:



- 1) Muestre una secuencia de operaciones que lleven al sistema a un estado de interbloqueo.
 - 2) Muestre como el interbloqueo anterior puede ser prevenido usando el algoritmo del banquero.
- c) Un proceso se encuentra en el estado "suspendido-espera".
- Indique la secuencia de transiciones por las cuales el proceso alcanza ese estado, partiendo de la llegada del mismo al sistema. Explique que eventos generan dichas transiciones.

- Muestre la secuencia de transiciones que pueden llevar al proceso a alcanzar el estado "listo".
 - b) Los tiempos de servicio (procesamiento) de tres procesos P1, P2 y P3 son 5ms, 3ms, y 10ms, respectivamente. Sean los períodos de $p1 = 25ms$ y $p2 = 8ms$, ¿Cuál es el valor más pequeño de $p3$ para que la planificación EDF pueda cumplir con los plazos de todos los procesos? Muestre gráficamente como sería dicha planificación
4. a) Explique la diferencia entre semáforos binarios, semáforos de conteo y mutex. ¿En qué circunstancia utilizaría cada uno?
- b) Considere las siguientes dos funciones, donde A y B son cálculos arbitrarios:

<pre>F1() { wait(s1); c1 = c1 + 1; if (c1 == 1) wait(d); signal(s1); A; wait(s1); c1 = c1 - 1; if (c1 = 0) signal(d); signal(s1); }</pre>	<pre>F2() { wait(s2); c2 = c2 + 1; if (c2 == 1) wait(d); signal(s2); B; wait(s2); c2 = c2 - 1; if (c2 = 0) signal(d); signal(s2); }</pre>
---	---

Inicialmente $s1=s2=d=1$ y $c1=c2=0$ Asuma que una cantidad no limitada de procesos invocan cualquiera de las funciones F1 o F2

- Cuántas invocaciones de A pueden proceder de forma concurrente? ¿Cuáles son los valores de $s1$, $c1$ y d en ese instante?
- Cuando A está corriendo, ¿Cuántas invocaciones de B pueden proceder concurrentemente? ¿Cuáles son los valores de $s2$, $c2$ y d en ese instante?
- ¿Pueden A o B sufrir inanición? explique porqué sí o porqué no.