

1 – INTRODUCCIÓN A LOS SISTEMAS OPERATIVOS

- 1.1 ¿Qué contiene una entrada de la tabla de vectores de interrupción?
- a) El nombre de la rutina de tratamiento.
 - b) La dirección de la rutina de tratamiento.
 - c) El número de la interrupción.
 - d) El nombre de la tarea del SO que trata la interrupción.
- 1.2 ¿Cuál de las siguientes instrucciones máquina no debería ejecutarse en modo protegido?
Razone su respuesta.
- a) Inhibir interrupciones.
 - b) Instrucción TRAP.
 - c) Modificar el reloj del sistema.
 - d) Cambiar el mapa de memoria.
- 1.3 ¿Cuáles son las principales funciones de un sistema operativo?
- 1.4 ¿Qué diferencia existe entre un mandato y una llamada al sistema?
- 1.5 ¿Que es la multiprogramación?
- 1.6 ¿Cuántas instrucciones de la siguiente lista deben ejecutarse exclusivamente en modo núcleo?
Razone su respuesta.
- Inhibir todas las interrupciones.
 - Escribir en los registros de control de un controlador de dma.
 - Leer el estado de un controlador de periférico.
 - Escribir en el reloj de la computadora.
 - Provocar un trap o interrupción software.
 - Escribir en los registros de la MMU.
- 1.7 ¿En qué época se introdujeron los primeros manejadores de dispositivos?
¿Y los sistemas operativos de tiempo compartido?
- 1.8 ¿Qué ventajas considera que tiene escribir un sistema operativo utilizando un lenguaje de alto nivel?
- 1.9 ¿El intérprete de mandatos de UNIX es interno o externo?
Razone su respuesta con un ejemplo
- 1.10 ¿Dónde es más compleja una llamada al sistema, en un sistema operativo monolítico o en uno por capas?
- 1.11 ¿Cuál de los siguientes mecanismos hardware no es un requisito para construir un sistema operativo multiprogramado con protección entre usuarios?
Razone su respuesta.
- a) Memoria virtual.
 - b) Protección de memoria.
 - c) Instrucciones de E/S que sólo pueden ejecutarse en modo kernel.
 - d) 2 modos de operación: kernel y usuario.
- 1.12 ¿Cómo se solicita una llamada al sistema operativo?
- 1.13 ¿Qué tipo de sistema operativo es más fácil de modificar, uno monolítico o uno por capas?
¿Cuál es más eficiente?
- 1.14 ¿Debe ser un sistema operativo multitarea de tiempo compartido?
¿Y viceversa? Razone su respuesta.

2 – ADMINISTRACION DE PROCESOS

- 2.1 Nombre las principales tareas que efectúa el SO en relación a la administración de Procesos
- 2.2 ¿Que es un proceso y que es un thread (hilo)? ¿En que se diferencian?
- 2.3 ¿Es un proceso un archivo ejecutable? Razone su respuesta.
- 2.4 En que estructura de datos guarda el SO datos referidos a un proceso. ¿Cuales son esos datos?
- 2.5 Que es multithreading? ¿Qué ventajas tiene con respecto a la multiprogramación?
- 2.6 Dibuje en un diagrama los 5 estados posibles de un proceso y enumere cuales son los eventos que causan las transiciones de un estado al otro.
- 2.7 Dibuje en un diagrama los 6 estados posibles de un proceso (considerando un estado suspendido) y enumere cuales son los eventos que causan las transiciones de un estado al otro.
- 2.8 Dibuje en un diagrama los 7 estados posibles de un proceso (considerando dos estados suspendido) y enumere cuales son los eventos que causan las transiciones de un estado al otro.
- 2.9 Indique cuál de estas operaciones no es ejecutada por el dispatcher (activador):
 - a) Restaurar los registros de usuario con los valores almacenados en la tabla del proceso.
 - b) Restaurar el contador de programa.
 - c) Restaurar el puntero que apunta a la tabla de páginas del proceso.
 - d) Restaurar la imagen de memoria de un proceso.
- 2.10 ¿Siempre se produce un cambio de contexto cuando se produce un cambio de proceso? Razone su respuesta.
- 2.11 ¿Cuál es la información que no comparten los procesos ligeros de un mismo proceso?

3 – PLANIFICACION DE PROCESOS

- 3.1 Nombre los distintos planificadores e indique cuales son sus funciones y donde se ubicarían dentro del diagrama de estados
- 3.2 ¿Puede producirse un cambio de contexto en un sistema con un planificador basado en el algoritmo SJF además de cuando se bloquea o se termina el proceso? Razone su respuesta.
- 3.3 ¿Qué algoritmo de planificación será más conveniente para optimizar el rendimiento de la CPU en un sistema que sólo tiene procesos en los cuales no hay entrada/salida?
- 3.4 ¿Cuál de las siguientes políticas de planificación es más adecuada para un sistema de tiempo compartido?
- SJF
 - Round-Robin
 - Prioridades
 - FIFO
- 3.5 ¿Cuál es el criterio de planificación más relevante en un sistema de tiempo compartido, el tiempo de respuesta o la optimización en el uso del procesador? Razone su respuesta.
- 3.6 ¿Cuál de las siguientes transiciones entre los estados de un proceso no se puede producir en un sistema con un algoritmo de planificación no expulsivo?
- Bloqueado a listo
 - Ejecutando a listo
 - Ejecutando a bloqueado
 - Listo a ejecutando
- 3.7 Sea un sistema que usa un algoritmo de planificación de procesos round-robin con un quantum de tiempo de 100 ms. En este sistema ejecutan dos procesos. El primero no realiza operaciones de E/S y el segundo solicita una operación de E/S cada 50 ms. ¿Cuál será el porcentaje de uso de la CPU?
- 3.8 Considere el siguiente conjunto de procesos planificados con un algoritmo round-robin con un quantum de 1, ¿Cuánto tardan en terminar todos ellos?

Proceso	Llegada	Duración CPU
P1	2	8
P2	0	5
P3	1	4
P4	3	3

- 3.9 En un sistema que usa un algoritmo de planificación de procesos round-robin, ¿cuántos procesos como máximo pueden cambiar de estado cuando se produce una interrupción del disco que indica que se ha terminado una operación sobre el mismo?

- 3.10 Se tienen los siguientes trabajos a ejecutar:

Proceso	Llegada	Duración CPU	Prioridad
P1	0	8	2
P2	1	5	4
P3	3	2	1
P4	4	7	3

La prioridad más alta es la de valor 1, se pide:

- Escribir un diagrama que ilustre la ejecución de estos trabajos usando:
 - Planificación de prioridades no expulsiva
 - Planificación de prioridades expulsiva
 - Planificación Round Robin con un quantum de tiempo de 2 unidades
 - FIFO
- Indicar cuál es el algoritmo de planificación con menor tiempo medio de espera

3.11 Considere el siguiente conjunto de procesos planificados:

Proceso	Llegada	Duración CPU
P1	1	8
P2	5	1
P3	2	6
P4	4	9
P5	3	4

- Indicar cuál es el algoritmo de planificación con menor tiempo medio de Retorno
 - Indicar cuál es el algoritmo de planificación con menor tiempo medio de Espera
- Considerando los siguientes algoritmos:

- SJF sin desalojo
- SJF con desalojo
- Round Robin con quantum de 3 unidades de tiempo

Realizar el Diagrama de Gantt correspondiente

3.12 Dados la siguiente tabla de procesos:

Proceso	Características	Instante de Llegada
P1	CPU 4, E/S 2, CPU 4	0
P2	CPU 3, E/S 2, CPU 2	1

Calcular el Tiempo Medio de Retorno, Tiempo medio de Espera y %uso de CPU para:

- SJF sin desalojo
- SJF con desalojo
- Round Robin con quantum de 2 unidades de tiempo

Realizar el Diagrama de Gantt correspondiente

3.13 Dados la siguiente tabla de procesos:

Proceso	Características	Instante de Llegada
P1	CPU 5, E/S 2, CPU 2	0
P2	CPU 1, E/S 3, CPU 5	1

Calcular el Tiempo Medio de Retorno, Tiempo medio de Espera y %uso de CPU para:

- SRT sin desalojo
- SRT con desalojo
- Round Robin con quantum de 2 unidades de tiempo

Realizar el Diagrama de Gantt correspondiente

3.14 Dados los siguientes datos

Proceso	Prioridad	Tiempo de Llegada	Tiempo de CPU
P1	0	4	3
P2	3	8	2
P3	1	6	8
P4	2	1	11
P5	2	3	9

Calcular:

- Tiempo Medio de Retorno
 - Tiempo Medio de Espera
- Considerando los siguientes algoritmos:
- FIFO
 - SJF sin desalojo
 - SJF con desalojo
 - SRT sin desalojo
 - SRT con desalojo
 - Prioridades sin desalojo. Considerar 0=mayor prioridad , 3=menor prioridad
 - Prioridades con desalojo

8) Round Robin con quantum de 2 unidades de tiempo

9) Round Robin con quantum de 5 unidades de tiempo

Realizar el Diagrama de Gantt correspondiente

3.15 Considere un sistema de multiprogramación que debe ejecutar dos procesos con las siguientes características:

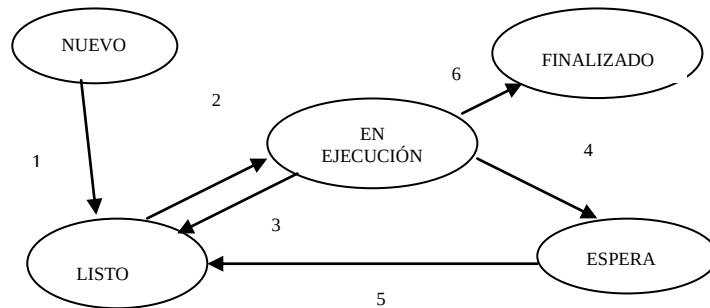
Proceso	Características
P1	CPU 40 ms, E/S disco 30 ms, CPU 10 ms, E/S cinta 20 ms, CPU 10 ms
P2	CPU 40 ms, E/S cinta 40 ms, CPU 30 ms, E/S disco 30 ms, CPU 10 ms

Considerar:

- Las interrupciones, incluido el overhead duran 10 mseg.
- El sistema cuenta con dos dispositivos de E/S, uno para unidad de cinta y otro para disco.

Se pide completar el diagrama que esta a continuación para los siguientes algoritmos:

- a) FIFO
- b) SJF sin desalojo
- c) SRT con desalojo
- d) Round Robin con quantum de 30 ms.



Tiempo	1	2	3	4	5	6	7	8	9	0	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
S.O.																																	
Proceso P1																																	
Proceso P2																																	
Interrupción																																	
Rutinas SO "P1"																																	
Rutinas SO "P2"																																	

3.16 Realice un diagrama de GANTT indicando el estado de cada proceso, sobre la base de las siguientes pautas y para los algoritmos indicados. Calcular el tiempo promedio de espera de los procesos, para los algoritmos: SJF con y sin desalojo (teniendo en cuenta la próxima ráfaga más corta) y RR (con un Quantum de 30 mseg.). El OVER HEAD es de 10 mseg. Primero llega el proceso A en el instante t_0 , después llega B (en $t_0 + 10$ mseg.) y por último llega C (en $t_0 + 20$ mseg.). Suponemos que las diferentes rutinas que se ejecutan no ocupan tiempo.

Proceso	Características
PA	CPU 20 ms., E/S disco 30ms, CPU 40 ms, E/S cinta 20 ms., CPU 10ms.
PB	CPU 30 ms., E/S disco 20 ms., CPU 40 ms.
PC	CPU 40 ms., E/S disco 30 mseg, CPU 20 ms.

SJF sin expropiación

Proceso/Tiempo	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34
Proceso A																																		
Proceso B																																		
Proceso C																																		
Interrupción	X	X	X																															
Rutinas SO "A"	2																																	
Rutinas SO "B"																																		
Rutinas SO "C"																																		

SJF con expropiación

Proceso/Tiempo	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34
Proceso A																																		
Proceso B																																		
Proceso C																																		
Interrupción	X	X	X																															
Rutinas SO "A"	2																																	
Rutinas SO "B"																																		
Rutinas SO "C"																																		

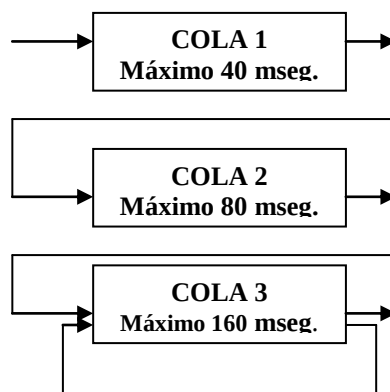
Round Robin

Proceso/Tiempo	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34
Proceso A																																		
Proceso B																																		
Proceso C																																		
Interrupción	X	X	X																															
Rutinas SO "A"	2																																	
Rutinas SO "B"																																		
Rutinas SO "C"																																		

3.17 Teniendo en cuenta el diagrama de estados que se detalla disponemos de los siguientes procesos para ejecutar en un sistema de planificación de colas múltiples con realimentación, los cuales llegan en el siguiente orden y deben efectuar las ráfagas que se indican:

- P1 260 mseg.,
- P2 130 mseg.,
- P3 240 mseg.,
- P4 30 mseg.,
- P5 20 mseg.,
- P6 210 mseg.,
- P7 80 mseg.,
- P8 40 mseg.,
- P9 120 mseg.

Se debe tener en cuenta que la cola 1 atiende 3 procesos seguidos, luego la 2 atiende 2 procesos seguidos y por último la 3 atiende 1 solo proceso. Indique en la siguiente tabla la secuencia en que se ejecutan los procesos



Facultad de Ingeniería

Proceso	Cola 1			Cola 2			Cola 3		
	Ejecuta	Resta	Rutina	Ejecuta	Resta	Rutina	Ejecuta	Resta	Rutina
P ₁									
P ₂									
P ₃									
P ₄									
P ₅									

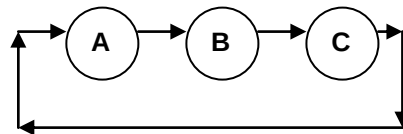
4 – ADMINISTRACION DE RECURSOS COMPARTIDOS, SINCRONIZACION, COMUNICACION ENTRE PROCESOS

- 4.1 ¿Qué significa el término espera ocupada?
- 4.2 ¿Qué consecuencia puede tener en el clock del sistema, deshabilitar las interrupciones?
- 4.3 ¿Por qué decimos que si las operaciones wait y signal no son ejecutadas atómicamente, la exclusión mutua puede ser violada?.
- 4.4 Dos procesos se comunican a través de un archivo, de forma que uno escribe en el archivo y el otro lee del mismo. Para sincronizarse, el proceso escritor envía una señal al lector. Proponga un esquema del código de ambos procesos. ¿Qué problema plantea la solución anterior?
- 4.5 El siguiente fragmento de código intenta resolver el problema de la sección crítica para dos procesos P0 y P1.
- ```

While (turno != i);
 SECCIÓN CRÍTICA;
 Turno = j;

```
- La variable turno tiene valor inicial 0. La variable i vale 0 en el proceso P0 y 1 en el proceso P1. La variable j vale 1 en el proceso P0 y 0 en el proceso P1. ¿Resuelve este código el problema de la sección crítica?
- 4.6 Indicar los semáforos necesarios para sincronizar los procesos y lograr las secuencias de ejecución pedidas a continuación, indicando sus valores iniciales, las operaciones wait y signal y el estado de cada semáforo en los distintos instantes de ejecución.

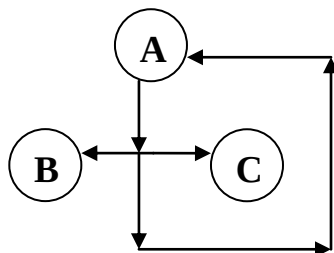
a) ABCABC...



| Semáforo      |         | SA | SB | SC |  |
|---------------|---------|----|----|----|--|
| Valor Inicial |         |    |    |    |  |
| Instante      | Proceso |    |    |    |  |
| T0            | A       |    |    |    |  |
| T1            | A       |    |    |    |  |
| T2            | B       |    |    |    |  |
| T3            | B       |    |    |    |  |
| T4            | C       |    |    |    |  |
| T5            | C       |    |    |    |  |

| Proc. A | Proc. B | Proc. C |
|---------|---------|---------|
|         |         |         |
|         |         |         |
|         |         |         |
|         |         |         |
|         |         |         |
|         |         |         |
|         |         |         |
|         |         |         |
|         |         |         |

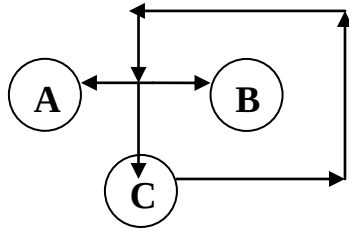
b) A(BoC)A(BoC)...



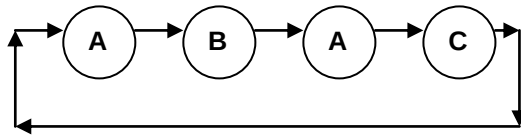


## Facultad de Ingeniería

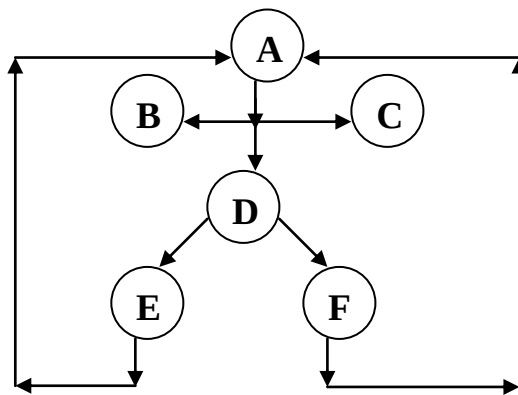
c) (AoB)C(AoB)C...



d) ABACABAC...



e) A(BoC)D(EyF)....



f) A B C B A B C B A B C B ..... (Ejercicio tomado en un parcial)

Sa = 1 , Sb = 0 , Sc = 0 , Sr = 1

| A     | B     | C     |
|-------|-------|-------|
| W(Sa) | W(Sb) | W(Sc) |
| W(Sr) |       | W(Sr) |
| S(Sb) | S(Sr) | S(Sb) |
| S(Sc) |       | S(Sa) |

**5 – ABRAZO MORTAL**

5.1 ¿Qué significa el término Abrazo Mortal?

5.2 Realizar la representación de la siguiente situación:

$N = \{ P_1, P_2, P_3, R_1(2), R_2(2), R_3(1) \}$

$A = \{ (R_1, P_1), (R_1, P_2), (R_2, P_2), (R_2, P_3), (P_1, R_2), (P_3, R_1) \}$

- A través del GAR
- Con la Representación matricial
- ¿Está el sistema en abrazo mortal? ¿Por qué?

5.3 Realizar la representación de la siguiente situación:

$N = \{ P_1, P_2, P_3, P_4, R_1(1), R_2(1), R_3(1) \}$

$A = \{ (R_1, P_1), (R_3, P_4), (R_2, P_2), (P_2, R_1), (P_3, R_2), (P_1, R_2) \}$

- A través del GAR
- Con la Representación matricial
- ¿Está el sistema en abrazo mortal? ¿Por qué?

5.4 Identifique por qué no es coherente el siguiente estado de asignación de recursos:

TOTAL DE RECURSOS:       $R_1 \rightarrow 2$        $R_2 \rightarrow 3$        $R_3 \rightarrow 2$

| Asignación |    |    |    | Requerido |    |    |    | Disponible |    |    |    |
|------------|----|----|----|-----------|----|----|----|------------|----|----|----|
|            | R1 | R2 | R3 |           | R1 | R2 | R3 |            | R1 | R2 | R3 |
| P0         | 2  | 0  | 0  | P0        | 0  | 1  | 2  | P0         | 0  | 0  | 2  |
| P1         | 0  | 2  | 0  | P1        | 1  | 0  | 0  | P1         |    |    |    |
| P2         | 0  | 2  | 0  | P2        | 0  | 0  | 0  | P2         |    |    |    |

5.5 Identifique por qué no es coherente el siguiente estado de asignación de recursos:

| Asignación |    |    |    | Máximo |    |    |    | Necesidad |    |    |    | Disponible |    |    |    |
|------------|----|----|----|--------|----|----|----|-----------|----|----|----|------------|----|----|----|
|            | R1 | R2 | R3 |        | R1 | R2 | R3 |           | R1 | R2 | R3 |            | R1 | R2 | R3 |
| P1         | 0  | 2  | 0  | P1     | 0  | 3  | 2  | P1        | 0  | 1  | 0  | P1         | 1  | 0  | 2  |
| P2         | 0  | 2  | 0  | P2     | 1  | 2  | 0  | P2        | 1  | 0  | 0  | P2         |    |    |    |

5.6 De acuerdo a las siguientes estructuras, supongamos que,  $C_{ij} + R_{ij} > E_j$  para algún  $i$ . ¿Cuáles son las implicaciones de esto para todos los procesos?

| Recursos en existencia<br>( $E_1, E_2, E_3, \dots, E_m$ ) |          |          |          |          | Recursos disponibles<br>( $D_1, D_2, D_3, \dots, D_m$ ) |          |          |          |          |
|-----------------------------------------------------------|----------|----------|----------|----------|---------------------------------------------------------|----------|----------|----------|----------|
| Matriz de asignación actual                               |          |          |          |          | Matriz de Necesidad                                     |          |          |          |          |
| $C_{11}$                                                  | $C_{12}$ | $C_{13}$ | $\dots$  | $C_{1m}$ | $R_{11}$                                                | $R_{12}$ | $R_{13}$ | $\dots$  | $R_{1m}$ |
| $C_{21}$                                                  | $C_{22}$ | $C_{23}$ | $\dots$  | $C_{2m}$ | $R_{21}$                                                | $R_{22}$ | $R_{23}$ | $\dots$  | $R_{2m}$ |
| $\vdots$                                                  | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$                                                | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| $\vdots$                                                  | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$                                                | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| $\vdots$                                                  | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$                                                | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| $C_{n1}$                                                  | $C_{n2}$ | $C_{n3}$ | $\dots$  | $C_{nm}$ | $R_{n1}$                                                | $R_{n2}$ | $R_{n3}$ | $\dots$  | $R_{nm}$ |

5.7 Considere la siguiente representación matricial de un sistema:

| Asignación |    |    |    |    | Máximo |    |    |    |    | Necesidad |    |    |    |    | Disponible |    |    |    |    |
|------------|----|----|----|----|--------|----|----|----|----|-----------|----|----|----|----|------------|----|----|----|----|
|            | R1 | R2 | R3 | R4 |        | R1 | R2 | R3 | R4 |           | R1 | R2 | R3 | R4 |            | R1 | R2 | R3 | R4 |
| P1         | 0  | 0  | 1  | 2  | P1     | 0  | 0  | 1  | 2  | P1        |    |    |    |    | P1         | 1  | 5  | 2  | 0  |
| P2         | 1  | 0  | 0  | 0  | P2     | 1  | 7  | 5  | 0  | P2        |    |    |    |    | P2         |    |    |    |    |
| P3         | 1  | 3  | 5  | 4  | P3     | 2  | 3  | 5  | 6  | P3        |    |    |    |    | P3         |    |    |    |    |
| P4         | 0  | 6  | 3  | 2  | P4     | 0  | 6  | 5  | 2  | P4        |    |    |    |    | P4         |    |    |    |    |
| P5         | 0  | 0  | 1  | 4  | P5     | 0  | 6  | 5  | 6  | P5        |    |    |    |    | P5         |    |    |    |    |

## Facultad de Ingeniería

- Determinar la matriz necesidad
- ¿El sistema esta en estado seguro? ¿Por qué?
- Si el proceso 2 requiere (0,4,2,0), ¿puede ser satisfecho sin pasar a un estado inseguro?
- Si P1 pide (0, 2, 4, 0), se puede atender de inmediato la solicitud

5.8 Considere la siguiente representación matricial de un sistema:

Existentes

| Asignados |    |    |    |    | Necesarios |    |    |    |    |   |    |    |    |    |
|-----------|----|----|----|----|------------|----|----|----|----|---|----|----|----|----|
|           | R1 | R2 | R3 | R4 |            | R1 | R2 | R3 | R4 |   | R1 | R2 | R3 | R4 |
| A         | 3  | 0  | 1  | 1  | A          | 1  | 1  | 0  | 0  | A | 6  | 3  | 4  | 2  |
| B         | 0  | 1  | 0  | 0  | B          | 0  | 1  | 1  | 2  | B |    |    |    |    |
| C         | 1  | 1  | 1  | 0  | C          | 3  | 1  | 0  | 0  | C |    |    |    |    |
| D         | 1  | 1  | 0  | 1  | D          | 0  | 0  | 1  | 0  | D |    |    |    |    |
| E         | 0  | 0  | 0  | 0  | E          | 2  | 1  | 1  | 0  | E |    |    |    |    |

Busque tres ordenamientos de estado seguro y si encuentra alguno inseguro indique porque es inseguro.

5.9 Dada las siguientes matrices determine cual de los ordenamientos dados es seguro y si encuentra alguno inseguro indique porque es inseguro.

TOTAL DE RECURSOS: A → 10      B → 5      C → 7

|                | Asignación |   |   | Máximo |   |   | Necesidad |   |   | Disponible |   |   |
|----------------|------------|---|---|--------|---|---|-----------|---|---|------------|---|---|
|                | A          | B | C | A      | B | C | A         | B | C | A          | B | C |
| P <sub>0</sub> | 0          | 1 | 0 | 7      | 5 | 3 |           |   |   |            |   |   |
| P <sub>1</sub> | 2          | 0 | 0 | 3      | 2 | 2 |           |   |   |            |   |   |
| P <sub>2</sub> | 3          | 0 | 2 | 9      | 0 | 2 |           |   |   |            |   |   |
| P <sub>3</sub> | 2          | 1 | 1 | 2      | 2 | 2 |           |   |   |            |   |   |
| P <sub>4</sub> | 0          | 0 | 2 | 4      | 3 | 3 |           |   |   |            |   |   |

|                | Asignación |   |   | Máximo |   |   | Necesidad |   |   | Disponible |   |   |
|----------------|------------|---|---|--------|---|---|-----------|---|---|------------|---|---|
|                | A          | B | C | A      | B | C | A         | B | C | A          | B | C |
| P <sub>1</sub> | 2          | 0 | 0 | 3      | 2 | 2 |           |   |   |            |   |   |
| P <sub>3</sub> | 2          | 1 | 1 | 2      | 2 | 2 |           |   |   |            |   |   |
| P <sub>4</sub> | 0          | 0 | 2 | 4      | 3 | 3 |           |   |   |            |   |   |
| P <sub>2</sub> | 3          | 0 | 2 | 9      | 0 | 2 |           |   |   |            |   |   |
| P <sub>0</sub> | 0          | 1 | 0 | 7      | 5 | 3 |           |   |   |            |   |   |

|                | Asignación |   |   | Máximo |   |   | Necesidad |   |   | Disponible |   |   |
|----------------|------------|---|---|--------|---|---|-----------|---|---|------------|---|---|
|                | A          | B | C | A      | B | C | A         | B | C | A          | B | C |
| P <sub>1</sub> | 3          | 0 | 2 | 3      | 2 | 2 |           |   |   |            |   |   |
| P <sub>3</sub> | 2          | 1 | 1 | 2      | 2 | 2 |           |   |   |            |   |   |
| P <sub>4</sub> | 0          | 0 | 2 | 4      | 3 | 3 |           |   |   |            |   |   |
| P <sub>0</sub> | 0          | 1 | 0 | 7      | 5 | 3 |           |   |   |            |   |   |
| P <sub>2</sub> | 3          | 0 | 2 | 9      | 0 | 2 |           |   |   |            |   |   |

5.10 Supongamos que el proceso A de la figura 6-10 solicita la última unidad de cinta. ¿Conduce esto a un bloqueo?

|   | UdeC | Plot | Impre | CD-R |
|---|------|------|-------|------|
| A | 3    | 0    | 1     | 1    |
| B | 0    | 1    | 0     | 0    |
| C | 1    | 1    | 1     | 0    |
| D | 1    | 1    | 0     | 1    |
| E | 0    | 0    | 0     | 0    |

Recursos Asignados

|   | UdeC | Plot | Impre | CD-R |
|---|------|------|-------|------|
| A | 1    | 1    | 0     | 0    |
| B | 0    | 1    | 1     | 2    |
| C | 3    | 1    | 0     | 0    |
| D | 0    | 0    | 1     | 0    |
| E | 2    | 1    | 1     | 0    |

Recursos necesarios

Existentes = E = (6, 3, 4, 2)

- 5.11 Una computadora tiene seis unidades de cinta, con n procesos en competencia por ellas. Cada proceso necesita dos unidades. ¿Para cuáles valores de n se libera el bloqueo del sistema?
- 5.12 Dada la siguiente matriz de asignación, de máximos requeridos y el vector de disponibles determinar si el sistema se encuentra en estado seguro.

| Máximo |    |    |    |    | Asignación |    |    |    |    | Disponibles |    |    |    | Necesidad |    |    |    |    |
|--------|----|----|----|----|------------|----|----|----|----|-------------|----|----|----|-----------|----|----|----|----|
|        | R1 | R2 | R3 | R4 |            | R1 | R2 | R3 | R4 | R1          | R2 | R3 | R4 |           | R1 | R2 | R3 | R4 |
| P1     | 4  | 1  | 1  | 1  | P1         | 3  | 0  | 1  | 1  | 1           | 1  | 1  | 1  | P1        |    |    |    |    |
| P2     | 0  | 2  | 1  | 2  | P2         | 0  | 1  | 0  | 0  |             |    |    |    | P2        |    |    |    |    |
| P3     | 4  | 2  | 1  | 0  | P3         | 1  | 1  | 1  | 0  |             |    |    |    | P3        |    |    |    |    |
| P4     | 1  | 1  | 1  | 1  | P4         | 1  | 1  | 0  | 1  |             |    |    |    | P4        |    |    |    |    |
| P5     | 2  | 1  | 1  | 0  | P5         | 0  | 0  | 0  | 0  |             |    |    |    | P5        |    |    |    |    |

- 5.13 Dada la siguiente matriz de asignación, y las distintas solicitudes en un instante dado, determinar si el sistema se encuentra en estado seguro.

Verificar si este ordenamiento es seguro en  $t_0$

|                | Asignación |   |   | Solicitud $t_0$ |   |   | Disponible |   |   |
|----------------|------------|---|---|-----------------|---|---|------------|---|---|
|                | A          | B | C | A               | B | C | A          | B | C |
|                |            |   |   |                 |   |   | 0          | 0 | 0 |
| P <sub>0</sub> | 0          | 1 | 0 | 0               | 0 | 0 |            |   |   |
| P <sub>1</sub> | 2          | 0 | 0 | 2               | 0 | 2 |            |   |   |
| P <sub>2</sub> | 3          | 0 | 3 | 0               | 0 | 0 |            |   |   |
| P <sub>3</sub> | 2          | 1 | 1 | 1               | 0 | 0 |            |   |   |
| P <sub>4</sub> | 0          | 0 | 2 | 0               | 0 | 2 |            |   |   |

Verificar si este ordenamiento es seguro en  $t_0$

|                | Asignación |   |   | Solicitud $t_0$ |   |   | Disponible |   |   |
|----------------|------------|---|---|-----------------|---|---|------------|---|---|
|                | A          | B | C | A               | B | C | A          | B | C |
|                |            |   |   |                 |   |   | 0          | 0 | 0 |
| P <sub>0</sub> | 0          | 1 | 0 | 0               | 0 | 0 |            |   |   |
| P <sub>2</sub> | 3          | 0 | 3 | 0               | 0 | 0 |            |   |   |
| P <sub>3</sub> | 2          | 1 | 1 | 1               | 0 | 0 |            |   |   |
| P <sub>1</sub> | 2          | 0 | 0 | 2               | 0 | 2 |            |   |   |
| P <sub>4</sub> | 0          | 0 | 2 | 0               | 0 | 2 |            |   |   |

Verificar si este ordenamiento es seguro en  $t_1$

|                | Asignación |   |   | Solicitud $t_1$ |   |   | Disponible |   |   |
|----------------|------------|---|---|-----------------|---|---|------------|---|---|
|                | A          | B | C | A               | B | C | A          | B | C |
|                |            |   |   |                 |   |   | 0          | 0 | 0 |
| P <sub>0</sub> | 0          | 1 | 0 | 0               | 0 | 0 |            |   |   |
| P <sub>1</sub> | 2          | 0 | 0 | 2               | 0 | 2 |            |   |   |
| P <sub>2</sub> | 3          | 0 | 3 | 0               | 0 | 1 |            |   |   |
| P <sub>3</sub> | 2          | 1 | 1 | 1               | 0 | 0 |            |   |   |
| P <sub>4</sub> | 0          | 0 | 2 | 0               | 0 | 2 |            |   |   |