



**Universidad de Las Palmas de Gran Canaria
Escuela Universitaria de Informática**

**Sistemas Operativos
Convocatoria de septiembre, año 2002
9 de septiembre de 2002**

Calificación
1
2
3
4
5
6

Nombre	Titulación

Dispone de tres horas para completar el examen.

1 (1,5 pts) Considere la siguiente carga de procesos:

Proceso	Tiempo de llegada	Duración de ciclo de procesador
A	0	3
B	1	5
C	3	2
D	9	5
E	12	5

Para las siguientes políticas: a) Round-Robin con cuanto igual a 4 y b) primero el más corto expulsivo. Obtenga el diagrama de Gantt y el tiempo de espera medio por cada política.

ROUND ROBIN (Q=4)

T0	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19
A	A	A	B	B	B	B	C	C	B	D	D	D	D	E	E	E	E	D	E

Tiempos de espera:

A	0
B	4
C	4
D	5
E	3

PRIMERO EL MAS CORTO EXPULSIVO

T0	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19
A	A	A	C	C	B	B	B	B	B	D	D	D	D	D	E	E	E	E	E

Tiempos de espera:

A	0
B	4
C	0
D	1
E	3

2 (1,75 pts) Tenemos un sistema de gestión de memoria paginada de **dos niveles** que utiliza una TLB. Se tienen los siguientes tiempos medios:

- Tiempo de acceso a la TLB: 4 nseg. (1 nseg = 10^{-9} seg.)
- Tiempo medio de acceso a memoria principal: 33 nseg.

Suponiendo que la TLB tiene una tasa del 98% de aciertos, ¿cuánto es el tiempo medio de acceso a memoria?

tiempo medio de acceso = $T_{acierto} * P_{acierto} + T_{fallo} * P_{fallo}$

$T_{acierto}$ = tiempo de acceso si hay acierto en TLB = $4 + 33 = 37$ nseg.

$P_{acierto}$ = probabilidad de acierto en TLB = 98%

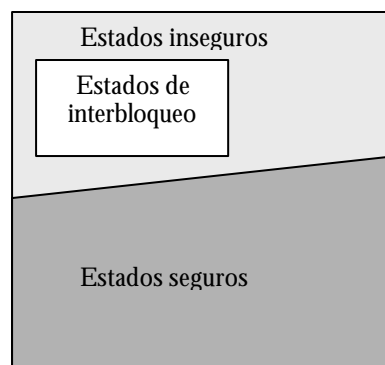
T_{fallo} = tiempo de acceso si falla la TLB = $4 + 33 + 33 + 33 = 103$ nseg.

P_{fallo} = probabilidad de fallo en TLB = 2%

El tiempo medio, por tanto, es: $0,98 * 37 + 0,02 * 103 = \mathbf{38,32}$ nseg.

3 (1,75 pts) Suponga un sistema que emplea el algoritmo del banquero para evitar caer en una situación de interbloqueo. En un instante dado y ante unas solicitudes de recursos, el sistema lanza el algoritmo y detecta que si concede los recursos se encontraría en un estado inseguro. Si, a pesar de esto, el sistema decide conceder los recursos solicitados, ¿puede afirmarse que el sistema se encuentra en interbloqueo?

No necesariamente. El conjunto de estados en los que podríamos afirmar que el sistema se encuentra en una situación de interbloqueo constituyen un subconjunto de los estados inseguros, es decir, un estado de interbloqueo es un estado inseguro pero no todos los estados inseguros son estados de interbloqueo (ver figura).



Nombre

4 (1,75 pts) Responda con claridad y brevedad a las siguientes cuestiones:

- a) ¿Qué ventaja aportan los manejadores de dispositivos de E/S en el diseño de un sistema operativo?

Comodidad, pues liberan al usuario programador de tener que incluir en su código el manejo de los dispositivos.

Seguridad, pues al formar parte del sistema operativo, es éste quien en última instancia realiza las operaciones de entrada salida.

- b) ¿Qué ventaja aportan los enlaces (*links*) en un sistema de archivos?

Flexibilidad en la organización lógica de los archivos, pues un mismo archivo puede ser referenciado con *pathname* diferentes.

- c) ¿En qué consiste el problema de las autoreferencias en los sistemas de archivos?

Un archivo posee referencias (*pathnames* distintos) que realmente ya no son válidas, éste fenómeno sólo se da en los directorios con estructura de grafo general.

- d) La mayoría de los sistemas actuales presentan dos modos de operación: modo privilegiado y modo no privilegiado. ¿Por qué? ¿Cuándo y cómo se cambia de modo?

Principalmente, porque en un sistema operativo multiprogramado o multiusuario existen restricciones de acceso a los recursos: protección de zonas de memoria, restricción del uso de archivos y dispositivos de E/S, etc. En segundo lugar, y muy relacionado con lo anterior, para poder preservar la integridad del sistema operativo, que no debe ser dañado por el uso indebido de los programas de usuario.

La CPU arranca en modo privilegiado. Cuando el SO cede el control al usuario, conmuta previamente a modo no privilegiado. Sólo se vuelve a modo privilegiado cuando el SO recupera el control, es decir, cuando ocurre una interrupción, una llamada al sistema o una excepción.

5 (1,5 pts) Dada esta cadena de referencias a memoria virtual:

8,1,3,1,4,5,2,3,4,5,1,2,3,6,1,8,4,1,3

Simule la gestión de estas referencias utilizando las siguientes políticas: a)LRU; b)segunda oportunidad. Suponga que el sistema dispone de tres marcos físicos que inicialmente están sin asignar.

	8	1	3	1	4	5	2	3	4	5	1	2	3	6	1	8	4	1	3
LRU	8	8	8	8	3	1	4	5	2	3	4	5	1	2	3	6	1	8	4
		1	1	3	1	4	5	2	3	4	5	1	2	3	6	1	8	4	1
			3	1	4	5	2	3	4	5	1	2	3	6	1	8	4	1	3
2ª OP.	→8 ₁	→8 ₁	→8 ₁	→8 ₁	4 ₁	4 ₁	→4 ₁	3 ₁	3 ₁	→3 ₁	1 ₁	1 ₁	→1 ₁	6 ₁	6 ₁	→6 ₁	4 ₁	4 ₁	→4 ₁
		1 ₁	1 ₁	1 ₁	→1 ₀	5 ₁	5 ₁	→5 ₀	4 ₁	4 ₁	→4 ₀	2 ₁	2 ₁	→2 ₁	1 ₁	1 ₁	→1 ₀	→1 ₁	1 ₀
			3 ₁	3 ₁	3 ₀	→3 ₀	2 ₁	2 ₀	→2 ₀	5 ₁	5 ₀	→5 ₀	3 ₁	3 ₁	→3 ₀	8 ₁	8 ₀	8 ₀	8 ₀

NOTAS segunda oportunidad:

- La flecha indica el apuntador al siguiente candidato a víctima.
- El número en subíndice es el bit de referencia de cada página.

6 (1,75 pts) Construya una solución al segundo problema de los lectores y escritores (prioridad para estos últimos), con el añadido de que se debe garantizar que nunca puede haber más de 10 lectores trabajando simultáneamente. Para su implementación puede optar por alguna de estas herramientas:

- semáforos
- monitores
- cerrojos y variables condición

Nombre

```
type Lectores_Escritores = monitor
var
  Lect, Esc: condition;
  nlec, nesc, nescbloq: entero;
  escribiendo: booleana;

procedure entry Empezar_Lectura()
begin
  nlec:=nlec+1;
  while nesc>0 or nlec>10 do
    Lect.Wait;
end;

procedure entry Terminar_Lectura()
begin
  nlec:=nlec-1;
  if nesc>0 and nlect=0 then
    Esc.Signal;
  else if nesc=0 then
    Lect.Signal;
  end if
end;

procedure entry Empezar_Escritura()
begin
  nesc:=nesc+1;
  while escribiendo or nlec>0 do
    nescbloq:=nescbloq+1;
    Esc.Wait;
    nescbloq:=nescbloq-1;
  end while;
  escribiendo:=verdadero;
end;

procedure entry Terminar_Escritura()
begin
  nesc:=nesc-1;
  if nescbloq>0 then
    Esc.Signal;
  else
    escribiendo:=falso;
    Lect.Broadcast;
  end if;
end;

begin
  nlec:=0;
  nesc:=0;
  nescbloq:=0;
  escribiendo:=falso;
end;

end.
```