

SISTEMAS OPERATIVOS

CERTAMEN #2

Pauta de corrección

Wenceslao Palma <wenceslao.palma@ucv.cl>

1. Responda las sgtes preguntas. Justifique cada una de sus respuestas.

(a) (7 ptos.) En un sistema que administra su memoria mediante paginación, los procesos tienen un espacio lógico de direccionamiento formado por 100 páginas. Cada una de las páginas tiene un tamaño de 2KB y la memoria física disponible es de 1MB. Determine el formato de la dirección lógica y física.

R.: Para direccionar 100 páginas se necesitan 7 bits. El desplazamiento está determinado por el tamaño de la página (2KB), luego se necesitan 11 bits. Por lo tanto, el formato de la dirección lógica es $7 + 11$ bits.

En el caso de la dirección física se debe determinar en cuántos frames se puede dividir la memoria física disponible (1MB) sabiendo que las páginas son de 2KB. Luego, se necesitan 9 bits ($2^{20}/2^{11} = 512$) para direccionar los frames y el desplazamiento queda igual que en el caso de la dirección lógica. Con todo, el formato de la dirección física es $9 + 11$ bits.

(b) (5 ptos.) Ante un fallo de página, se produce un cambio de contexto?

R.: sí, cuando el proceso se está ejecutando en modo usuario se necesitará pasar a modo kernel para cargar la página que falta.

(c) (8 ptos.) Considere una serie de referencias a páginas para un proceso con un conjunto residente de tamaño M inicialmente vacío. La serie de referencias a páginas es de longitud P con N números de página distintos en ella. Para cualquier algoritmo de reemplazo de páginas:

– Cuál es el menor número de fallos de página?

R.: 0, ocurre cuando $N = M$.

– Cuál es el mayor número de fallos de página?

R.: $P - M$, ocurre cuando luego de llenar el conjunto residente todas las referencias posteriores provocan un fallo de página.

2. (20 ptos.) Considere un sistema con memoria virtual donde el conjunto residente es de tamaño 4. Muestre el funcionamiento de los algoritmos de reemplazo de página OPT, FIFO, LRU y CLOCK para la siguiente secuencia de referencias a páginas:
1 2 3 4 1 2 5 1 2 3 4 5.
Comente.

OPT (4 ptos)

1	2	3	4	1	2	5	1	2	3	4	5
1	1	1	1	1	1	1	1	1	1	4	4
	2	2	2	2	2	2	2	2	2	2	2
		3	3	3	3	3	3	3	3	3	3
			4	4	4	5	5	5	5	5	5
						F				F	

FIFO (4 ptos)

1	2	3	4	1	2	5	1	2	3	4	5
1	1	1	1	1	1	5	5	5	5	4	4
	2	2	2	2	2	2	1	1	1	1	5
		3	3	3	3	3	3	2	2	2	2
			4	4	4	4	4	4	3	3	3
						F	F	F	F	F	F

LRU (5 ptos)

1	2	3	4	1	2	5	1	2	3	4	5
1	1	1	1	1	1	1	1	1	1	1	5
	2	2	2	2	2	2	2	2	2	2	2
		3	3	3	3	5	5	5	5	4	4
			4	4	4	4	4	4	3	3	3
						F			F	F	F

CLOCK (5 ptos)

1	2	3	4	1	2	5	1	2	3	4	5
*1-1	*1-1	*1-1	*1-1	*1-1	*1-1	5-1	5-1	5-1	*5-1	4-1	4-1
	2-1	2-1	2-1	2-1	2-1	*2-0	1-1	1-1	1-1	*1-0	5-1
		3-1	3-1	3-1	3-1	3-0	*3-0	2-1	2-1	2-0	*2-0
			4-1	4-1	4-1	4-0	4-0	*4-0	3-1	3-0	3-0
						F	F	F	F	F	F

3. (20 ptos.) El sgte código corresponde al proceso productor de una posible solución al problema del productor/consumidor. Es correcto dicho código? justifique.

```
semaforo s = 1;           // sección crítica
semaforo vacio = n;       // elementos disponibles en el buffer
semaforo lleno = 0;       // elementos ocupados en el buffer

while (true){
    wait(s);
    wait(vacio);
    producir();
    signal(lleno);
    signal(s);
}
```

R.: El código no es correcto.

El productor se bloqueará si ingresa en la sección crítica y no hay espacio en el buffer. Si el consumidor consume al menos un producto el productor se desbloquea pero esto no sucederá ya que el productor no ha salido de la sección crítica.

En este caso la solución es verificar si hay espacio en el buffer y luego ingresar en la sección crítica.