

Examen de Sistemas Operativos Avanzados

Apellidos, nombre _____

DNI: _____ Grado en Ingeniería Informática Computadores

Ejercicio 1

Cálculos justificativos:

❶

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
000-																
001-																
...														
010-																
011-																
012-																
013-																
...														

Cuadro 1: Mapa de memoria principal en el instante t.

❷ Marcos asignados

❸

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
000-																
001-																
...														
010-																
011-																
012-																
013-																
...														

Cuadro 2: Mapa de memoria principal en el instante t+1.

Marcos asignados:

④

▪ 0x0271

▪ 0x0371

▪ 0x0272

▪ 0x0372

▪ 0x0471

⑤

Cad. refs.	-	08							
A/F	-	F							
Marcos	-								
Páginas	..89ABC..								
bit R	..00000..								
Contador	..00000..								
Cad. refs.									
A/F									
Marcos									
Páginas									
bit R									
Contador									

Cuadro 3: Ejecución del algoritmo de reemplazo NFU.

Examen de Sistemas Operativos Avanzados

Ejercicios prácticos

Problema 1

Se dispone de un sistema operativo que se ejecuta en un computador con una unidad de gestión de memoria (MMU) que soporta paginación pura con las siguientes características:

- Direcciones físicas y virtuales de 16 bits y direccionamiento a nivel de byte.
- Tamaño de página y de marco de página de 256 bytes.
- Las tablas de páginas tienen un máximo de 128 entradas.
- El formato de una entrada en la tabla de páginas es: V (1 bit) - Número de marco (7 bits), donde V=1 indica que el marco es válido y está presente en memoria principal.
- El marco de página 0 se utiliza para guardar la tabla de Bloques de Control de Procesos (BCPs). Cada BCP ocupa 8 bytes y presenta la siguiente estructura:

```
struct BCP {
    unsigned char PID; /* 1 byte. Proceso existe si su pid distinto de 0*/
    unsigned char RLTP; /* 1 byte */
    short int RPBTP; /* 2 bytes */
    unsigned char extra[4]; /* 4 bytes */
}
```

- El marco de página 1 se reserva para almacenar las tablas de páginas de los procesos.

En este sistema se produce una determinada secuencia de eventos que gestiona el sistema operativo de la siguiente manera:

- Cuando se **inicia un proceso**:
 1. Asigna un BCP libre en el marco 0.
 2. Crea la tabla de páginas en el marco 1.
 3. Se procede a la carga del proceso en memoria, asignando los marcos necesarios.
- Cuando **finaliza un proceso**, el sistema pone a 0 toda la información gestionada por dicho proceso en los marcos 0 y 1.

Suponga que en el instante $t+0$ únicamente existen dos procesos en el sistema y cargados completamente en memoria principal: el proceso 1 requiere de 1 KB de memoria y el proceso 2 requiere de 2 KB.

1. Rellene la tabla adjunta indicando el contenido de los marcos 0 y 1 en el instante $t+0$. (Marque la información desconocida con 'X'). *(0,3 puntos)*
2. Indique qué marcos de memoria están ocupados en el instante $t+0$. *(0,2 puntos)*

RESPUESTA (apartado 1 y apartado 2):

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
000-	01	04	01	00	XX	XX	XX	XX	02	08	01	04	XX	XX	XX	XX
001-																
...														
010-	82	83	84	85	86	87	88	89	8A	8B	8C	8D				
011-																
...														

Cuadro 4: Mapa de memoria principal en el instante t.

Marcos de memoria ocupados:

0x00, 0x01, 0x02, 0x03, 0x04, 0x05, 0x06, 0x07, 0x08, 0x09, 0x0A, 0x0B, 0x0C y 0x0D.

En el instante t+1 ha finalizado el proceso 2 y se acaba de iniciar el proceso 3, que requiere 1.536 bytes (1,5 KB) de memoria.

3. Rellene la tabla indicando el contenido del marco 0 y del marco 1 en el instante t+1.

Indique los marcos de memoria que estarán ocupados en dicho instante. (0,5 puntos)

RESPUESTA:

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
000-	01	04	01	00	XX	XX	XX	XX	03	06	01	04	XX	XX	XX	XX
001-																
...														
010-	82	83	84	85	86	87	88	89	8A	8B	00	00				
011-																
...														

Cuadro 5: Mapa de memoria principal en el instante t.

Marcos de memoria ocupados:

0x00, 0x01, 0x02, 0x03, 0x04, 0x05, 0x06, 0x07, 0x08, 0x09, 0x0A y 0x0B.

En el instante $t+2$ el proceso 1 cambia su estado de listo a ejecución referenciando las siguientes direcciones de memoria:

0x0271, 0x0371, 0x0272, 0x0372, 0x0471

4. Realice la traducción de direcciones virtuales a físicas, sabiendo que la MMU de dicho computador dispone de un *buffer* de traducción adelantada (TLB) con dos únicas entradas, que contienen referencias realizadas anteriormente por otros procesos.

Indique en qué casos se obtiene la traducción a partir del TLB y en qué casos se obtiene a partir de la tabla de páginas. (0,5 puntos)

RESPUESTA:

A continuación se muestran los valores de los campos página (P) y desplazamiento (Desp), junto con la traducción a la dirección física de memoria principal. Se considera que en todos los casos el bit $V=1$.

0x0271 – P: 02, Desp: 71. Dirección física: 0x0471. La página no se encuentra en el TLB, se obtiene la traducción de la tabla de páginas y se actualiza el TLB.

0x0371 – P: 03, Desp: 71. Dirección física: 0x0571. La página no se encuentra en el TLB, se obtiene la traducción de la tabla de páginas y se actualiza el TLB.

0x0272 – P: 02, Desp: 72. Dirección física: 0x0472. La página se encuentra en el TLB.

0x0372 – P: 03, Desp: 72. Dirección física: 0x0572. La página se encuentra en el TLB.

0x0471 – P: 04, Desp: 71. Se exceden los límites de memoria del proceso, $RLTP = 0x04$, se genera una excepción.

En el instante $t+10$ se inicia el proceso 4, de 4 KB, al que se le han asignado tres marcos de página vacíos.

5. Simule el algoritmo de reemplazo *Not Frequently Used* (NFU) sobre la siguiente cadena de referencias y muestre los aciertos (A) y los fallos de página (F) que se producen. (0,5 puntos)

0x08 0x09 (INT) 0x08 0x0A 0x09 0x08 0x0B
(INT) 0x0A 0x0B 0x0A 0x0C (INT) 0x0A 0x0B 0x0C

Nota: INT indica cuándo produce la interrupción que activa el algoritmo de reemplazo.

RESPUESTA:

	P: 08	P: 09	INT	P: 08	P: 0A	P: 09	P: 08	P: 0B
A/F	F	F		A	F	A	A	F
Marco	08 - -	08 09 -		08 09 -	08 09 0A	08 09 0A	08 09 0A	08 09 0B
Págs.: 89ABC			89ABC					
bit R: 00000			11000					
Cont.: 00000			11000					

	INT	P: 0A	P: 0B	P: 0A	P: 0C	INT	P: 0A	P: 0B	P: 0C
A/F		F	F	F	F		F	F	F
Marco		08 09 0A	08 09 0B	08 09 0A	08 09 0C		08 09 0A	08 09 0B	08 09 0C
Págs.	89ABC					89ABC			
bit R	11110					00111			
Cont.	22110					22221			

Cuadro 6: Ejecución del algoritmo de reemplazo NFU.

(2 puntos)