

Problema de planificación de procesos y sincronización

2. (10 puntos) Se desea realizar un conjunto de pruebas en el sistema operativo *GeekS-quared* utilizando un nuevo algoritmo de planificación de hilos expulsivo (con requisa) y basado en prioridades, cuyas características se describen a continuación:

- Maneja 32 colas de hilos, una por cada prioridad. Si el valor de prioridad de un hilo es mayor que el de otro hilo, esto implica que el primero tiene mayor prioridad. Por ejemplo, un hilo con prioridad 20 tiene más prioridad que uno con prioridad 18.
- El algoritmo de planificación utilizado en cada cola es *round-robin* con un *quantum* de 5 u.t. El planificador siempre elegirá como siguiente hilo a ejecutarse el situado en la cabeza de la cola de mayor prioridad.
- Las prioridades de 16 a 31 se reservan para hilos de tiempo real y son **estáticas**.
- Las prioridades desde 0 a 15 se reservan para los hilos ordinarios. Estos hilos heredan en su creación la prioridad de la tarea a la que pertenecen (*prioridad base*). Durante la existencia de estos hilos, su prioridad está comprendida entre la *prioridad base* y 15 y se ajusta dinámicamente según los siguientes escenarios de planificación:
 - **El hilo agota su *quantum*:** Sólo si el valor de la prioridad del hilo es mayor que la *prioridad base*, el planificador disminuye en uno la prioridad del hilo. En cualquier caso, a continuación, el planificador coloca al hilo al final de la cola correspondiente a su prioridad.
 - **Un hilo transita del estado de *Bloqueado a Listo*:** Aumenta su prioridad en función del tipo de evento por el que espera. Por ejemplo, el *boosting* es de 1 u. t. sobre la *prioridad actual* cuando la espera es por un semáforo.
 - **Requisa por la llegada al sistema de un hilo de mayor prioridad:** El hilo que se está ejecutando es expulsado de la CPU y pasa a la cabeza de la cola correspondiente a su prioridad **reiniciándose a 0 el valor de su *quantum***. El nuevo hilo, de mayor prioridad, pasa a ejecutarse.

En este sistema existe en $t=5$ únicamente un hilo principal, *Hp*, con prioridad 5, y tres hilos creados por él en este instante: *H10*, *H00* y *H01*. Los hilos *H00* y *H01* tienen asociada en su creación la función *Alumno()* y el hilo *H10* la función *Profesor()*, respectivamente.

A continuación, se muestran las funciones ejecutadas por los hilos hijos definidas a nivel de diseño:

Alumno()

```
P(exmut);  
if (esperando < MAX) {  
    esperando = esperando + 1;  
    V(alumnos);  
    V(exmut);  
    P(profe);  
    Recibir_Tutoria(); /* 4 u.t. */  
} else {  
    V(exmut);  
}
```

Profesor()

```
while(TRUE) {  
    P(alumnos);  
    P(exmut);  
    esperando = esperando - 1;  
    V(profe);  
    V(exmut);  
    Realizar_Tutoria(); /* 4 u. t. */  
}
```

Considere para este problema los siguientes supuestos:

- MAX y TRUE son constantes; MAX definida con valor 10 y TRUE con valor 1.
- Los semáforos son binarios, compartidos por todos los hilos, y gestionan una cola FIFO de hilos bloqueados por ellos.
- Realizar_Tutoria() y Recibir_Tutoria() son invocaciones a dos funciones. Ambas suponen *4 u. t. de CPU*.
- Por simplicidad, operaciones como las siguientes: operaciones aritméticas, de comparación, de manejo de semáforos -independientemente de su comportamiento en cada escenario concreto-, consumen 1 u.t. de CPU. Por lo tanto, observe que en la ejecución de una sentencia **if-else**, como la que aparece en las función Alumno(), debe considerar 1 u.t. por la evaluación de su condición, además de las u.t. asociadas a la ejecución de cada sentencia del bloque **if/else**, según el resultado de la evaluación de la condición. De forma similar se supondrá para la sentencia **while**¹.

A continuación, responda a las siguientes preguntas:

- (a) (7 puntos) Complete la plantilla adjunta partiendo del estado inicial del sistema que aparece en la unidad de tiempo $t=5$. Tenga en cuenta además que el orden que ocupan los hilos hijos en la cola de prioridad 5 en $t=5$ coincide con su **orden de creación: H10, H00 y H01**, y que en $t=6$, el hilo principal **Hp** queda en espera **inmediatamente** por la finalización de sus hilos hijos, en el mismo orden en que éstos se crearon. Rellene en el diagrama de la hoja adjunta: el estado en el que se encuentra cada hilo en cada instante (*Ejecución, Listo o Bloqueado*), el valor que toman los tres semáforos implicados en este escenario, el valor la variable compartida **esperando** y el estado de las colas asociadas a los semáforos, desde la u.t. $t=6$ hasta $t=32$.

¹1 u. t. por la evaluación de su condición y 1 u. t. por cada operación ejecutada en el bloque del **while**, excepto Realizar_Tutoria(), que supone 4 u.t. de CPU.

Solución: Se adjunta la plantilla al final de la solución.

(b) (3 puntos) Indique el instante de tiempo en el que se producen por primera vez los siguientes eventos:

- Desalojo o requisa.
- Cesión voluntaria de la CPU.
- Finalización de *quantum*.

Solución:

- *Expulsión o desalojo. En $t=12$, por la llegada del hilo H10, de mayor prioridad, tras su bloqueo en la cola asociada al semáforo **alumnos**.*
- *Cesión voluntaria de la CPU. En $t=6$ el hilo Hp, al quedarse bloqueado en espera de sus tres hilos hijos.*
- *Finalización de quantum. El primer quantum que finaliza se produce en $t=19$ para el hilo H10; este hilo es el de mayor prioridad en esa unidad de tiempo por lo que se reinicia su quantum a 0 y sigue ejecutándose.*

t_i	0	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	
H_{00}						L			E	E	E	E	L										E	E	E	E	E							
Pr. Cola					5																													
H_{01}					L																													
Pr. Cola					5																								E	E	E	E	L	
H_{10}					L	E	E	B					E	B	E	E	E	E	E	E	E	E	B										E	
Pr. Cola					5								6		7					6													7	
H_p				E	E	B																												
Pr. Cola				5																														
exmut	1									0								1												0				
alumnos	0																																	
profe	0																1								0									
esperando	0											1				0																1		
Cola														H_{10}	–																			
exmut																																		
Cola									H_{10}				–											H_{10}										–
alumnos																																		
Cola																																		
profe																																		

Leyenda:

- E** Ejecución.
- L** Listo.
- B** Bloqueado.