

| | |
|--|--|
| Material permitido: Solo calculadora no programable | Aviso 1: Todas las respuestas deben estar debidamente razonadas. |
| Tiempo: 120 minutos | Aviso 2: Escriba con buena letra y evite los tachones. |
| N1 | Aviso 3: Solución del examen y fecha de revisión en http://www.uned.es/71902048/ |

1. Explique **razonadamente** si las siguientes afirmaciones son verdaderas o falsas:

- I) (1 p) La única causa que origina la creación de un proceso por parte del sistema operativo es que el proceso en ejecución invoque a una llamada al sistema para crear otro proceso.
 - II) (1 p) El planificador a medio plazo de un sistema operativo se encarga de decidir que proceso en la cola de procesos preparados será ejecutado a medio plazo en el procesador.
 - III) (1 p) El manipulador de la interrupción del reloj del computador únicamente se encarga de desbloquear al driver del reloj.
 - IV) (1 p) El tamaño óptimo de bloque de disco debe ser un compromiso entre el espacio desperdiciado en el disco y el tiempo de transferencia del bloque.
2. (2 p) Describir el funcionamiento de las operaciones `wait_sem` y `signal_sem` cuando se aplican sobre un semáforo general y cuando se aplican sobre un semáforo binario.
3. (2 p) Supóngase un sistema formado por tres procesos A , B y C y los siguientes recursos: dos lectores de CDs R_1 , una impresora R_2 y un disco duro R_3 . Dada la siguiente situación:
- El proceso A posee uno de los recursos R_1 .
 - El proceso B posee uno de los recursos R_1 y el recurso R_2 , y solicita el recurso R_3 .
 - El proceso C posee el recurso R_3 y solicita un recurso R_1 .

Dibujar el grafo de asignación de recursos y analizar **razonadamente** la posible existencia de interbloqueo.

Material permitido: **Solo calculadora no programable**Tiempo: **120 minutos**

N1

Aviso 1: Todas las respuestas deben estar debidamente razonadas.**Aviso 2:** Escriba con buena letra y evite los tachones.**Aviso 3:** Solución del examen y fecha de revisión en <http://www.uned.es/71902048/>

4. (2 p) En un computador con una capacidad de memoria principal de 64 KiB y palabras de 1 byte se utiliza gestión de memoria mediante segmentación. La tabla de segmentos (todos los datos numéricos están en decimal) es la siguiente:

| Nº de segmento | Base | Longitud |
|----------------|-------|----------|
| 0 | 0 | 7230 |
| 1 | 16384 | 8191 |
| 2 | 32768 | 1024 |
| 3 | 8192 | 356 |
| 4 | 24576 | 4200 |

Se pide:

- (1 p) Supuesto que una dirección lógica tiene el mismo tamaño en bits que una dirección física y que consta de los campos [nº de segmento, desplazamiento], determinar el tamaño en bits de cada uno de estos campos.
- (1 p) Determinar, si es posible, a qué direcciones físicas expresadas en decimal corresponden las siguientes direcciones lógicas expresadas en decimal: i) (0,4526), ii) (3,400).

SISTEMAS OPERATIVOS (Cód. 71902048)

Solución Examen Enero 2022

Solución Ejercicio 1

- I) Esta afirmación es **FALSA** ya que existen diferentes causas, no solo una, que originan la creación de un proceso por parte del sistema operativo, algunas de las más importantes son las siguientes:
- *Arranque del sistema operativo.* Al arrancar el sistema operativo se crean diversos procesos. Algunos son procesos que se ejecutan en segundo plano, como es el caso de los procesos demonio y los procesos del sistema. Otros son procesos que se ejecutan en primer plano, como por ejemplo, los intérpretes de comandos y los gestores de ventanas.
 - *Interacción del usuario con un intérprete de comandos o un entorno de ventanas.* En sistemas operativos interactivos, cuando un usuario teclea una orden en un intérprete de comandos o hace clic con el ratón sobre un icono de una ventana se crea un nuevo proceso para atender la petición del usuario.
 - *Inicio de un trabajo por lotes.* Cada vez que se inicia un trabajo por lotes se crea un proceso asociado a la ejecución de dicho trabajo.
 - *Un proceso en ejecución invoca a una llamada al sistema para crear otro proceso.* Una determinada tarea puede plantearse para ser realizada en varias fases independientes, cada una de las cuales puede ser realizada por un proceso. Así un proceso puede realizar llamadas al sistema para solicitar la creación de otros procesos que le ayuden en la realización de una tarea.
- II) El *planificador a medio plazo*, también conocido como *planificador de memoria* o *intercambiador*, es el componente del sistema operativo que se encarga de decidir qué procesos en memoria principal de los pertenecientes a la cola de procesos en el estado preparado o la cola de procesos en el estado bloqueado serán intercambiados a memoria secundaria, pasando a formar parte de la cola de procesos preparados en memoria secundaria o de la cola de procesos bloqueados en memoria secundaria, respectivamente. También se encarga de realizar la decisión contraria, es decir, qué procesos en las colas de memoria secundaria pasarán a las colas de memoria principal. En conclusión la afirmación es **FALSA**.
- III) El sistema operativo utiliza el driver del reloj para interactuar con el reloj del computador. Cuando el reloj genera una interrupción, el manipulador de la interrupción desbloquea el driver y éste debe realizar diferentes tareas, como por ejemplo: decrementar el cuanto del proceso en ejecución, incrementar el tiempo de uso del procesador de dicho proceso, incrementar el reloj de tiempo real y decrementar el tiempo de disparo en la lista de alarmas (y en la lista de tareas periódicas). Dependiendo de cada sistema operativo, algunas de las tareas anteriormente comentadas que debe realizar el driver del reloj son delegadas en el manipulador de la interrupción de reloj. En conclusión la afirmación es **FALSA**.
- IV) Un aspecto que afecta a la eficiencia del sistema es el tamaño del bloque de disco que se va a utilizar. Si el tamaño de bloque es muy pequeño un archivo puede ocupar muchos bloques, lo que aumentará el número de accesos al disco. Sin embargo la fragmentación interna dentro de un bloque será muy pequeña con lo que se reduce el desperdicio de espacio. Por otra parte si el tamaño de bloque es muy grande un archivo constará de pocos bloques y se reduce el número de accesos, pero por el contrario la fragmentación interna será grande desperdiciando mucho espacio. En consecuencia, el tamaño de bloque debe ser un compromiso entre el espacio en disco desperdiciado

y el número de bloques que se requieren para almacenar un archivo. Diversos estudios han puesto de manifiesto que el tamaño óptimo de bloque debe ser de unos pocos KiB. En conclusión la afirmación es **FALSA**.

Solución Ejercicio 2

El funcionamiento de las operaciones `wait_sem` y `signal_sem` cuando se aplican sobre un *semáforo general* S es el siguiente:

- `wait_sem(S)`. Esta operación disminuye en una unidad el valor del semáforo, es decir, $S=S-1$. Si el valor resultante es un número negativo entonces el proceso que ha invocado esta operación es añadido a la cola de procesos bloqueados asociada al semáforo y se bloquea.
- `signal_sem(S)`. Esta operación incrementa en una unidad el valor del semáforo, es decir, $S=S+1$. Si el valor resultante es menor o igual a 0 entonces se elimina de la cola asociada al semáforo uno de los procesos bloqueados, y se le despierta, lo que hace que pase al estado preparado para ejecución.

En el caso de un *semáforo binario* S las operaciones `wait_sem` y `signal_sem` funcionan de la siguiente manera:

- `wait_sem(S)`. Esta operación comprueba el valor del semáforo. Si $S=0$, entonces el proceso es colocado en la cola de procesos bloqueados asociada al semáforo y se bloquea. Si $S=1$, entonces pone el semáforo a 0 y el proceso puede continuar su ejecución.
- `signal_sem(S)`. Esta operación comprueba si la cola de procesos bloqueados asociada al semáforo S está vacía. En caso afirmativo, pone el semáforo a 1, y continúa su ejecución. En caso negativo, es decir, hay procesos bloqueados en el semáforo, entonces el sistema operativo elimina de la cola asociada al semáforo a uno de los procesos bloqueados, y le despierta lo que hace que pase al estado preparado para ejecución.

Solución Ejercicio 3

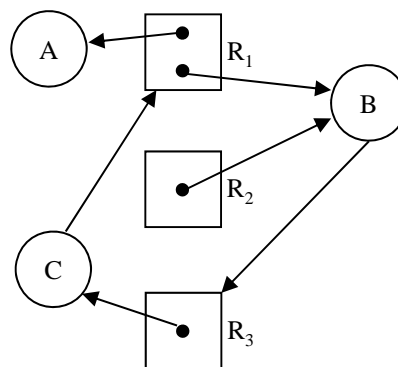


Figura 1

En un grafo de asignación de recursos se representa a cada proceso con un círculo y a cada recurso con un cuadrado. Cada instancia de un mismo recurso, se representa como un punto dentro del cuadrado asignado al recurso. Si un proceso ha solicitado una instancia de recurso y se encuentra bloqueado en espera de que le sea asignada, entonces se representa una flecha que sale del proceso hacia el recurso. Por otra parte, si un proceso tiene asignado una instancia de un recurso, entonces se representa una flecha

que va desde la instancia del recurso hacia el proceso. En la Figura 1 se representa el grafo de asignación de recursos pedido.

El grafo de asignación de recursos puede utilizarse para detectar la presencia de interbloqueos, para ello debe suponerse que ya se cumplen las condiciones de exclusión mutua y no existencia de expropiación, las cuales quedan fijadas por el sistema operativo. Se puede demostrar que si el grafo no contiene ningún camino que sea un ciclo entonces no existe interbloqueo. Por otra parte la existencia de un ciclo no es condición suficiente para que exista interbloqueo, dependerá también del número de instancias de cada recurso implicado en el ciclo y del tipo de caminos de los que formen parte dichas instancias. Si los recursos que forman parte de un ciclo únicamente tienen una instancia entonces existe interbloqueo. Por otra parte, si en el ciclo existen recursos que tienen más de una instancia, entonces para que exista interbloqueo todos las instancias de dichos recursos deben formar parte de caminos que sean ciclos.

En la Figura 1 se observa que existe el ciclo $R_1 \rightarrow B \rightarrow R_3 \rightarrow C \rightarrow R_1$, es decir, el proceso B tiene asignada una instancia del recurso R_1 y se encuentra bloqueado en espera de que le asignen R_3 . Por su parte, el proceso C tiene asignado R_3 y se encuentra bloqueado en espera de que le asignen una instancia de R_1 . Sin embargo, **no existe interbloqueo** ya que el otro camino que sale del nodo asociado al recurso R_1 , es decir, $R_1 \rightarrow A$, no forma un ciclo. Cuando el proceso A termine de usar la instancia del recurso R_1 que posee se la podrá asignar al proceso C. A su vez el proceso C, una vez consiga una instancia del recurso R_1 , puede liberar R_3 , permitiendo así que el proceso B pueda acceder a dicho recurso.

Solución Ejercicio 4

- a) Supuesto que la unidad direccionable es la palabra el tamaño n de una dirección de memoria física se puede obtener del dato de la capacidad de la memoria principal:

$$C_{MP} = 64 \text{ kibipalabras} = 2^{16} \text{ palabras.}$$

Conocido C_{MP} , el tamaño de este campo se obtiene resolviendo la siguiente desigualdad:

$$\min_f \{C_{MP} \leq 2^n\}$$

Luego como $C_{MP} = 2^{16}$ palabras entonces $n = 16$ bits.

Por otra parte en la tabla de segmentos se observa que la memoria principal está dividida en $N_S = 5$ segmentos. Conocido N_S , el tamaño de este campo se obtiene resolviendo la siguiente desigualdad:

$$\min_f \{N_S \leq 2^s\}$$

Luego como $N_S = 5$ segmentos entonces $s = 3$ bits.

Luego el tamaño del campo número de segmento de una dirección lógica de cualquiera de estos cinco segmentos tendrá un tamaño de $s = 3$ bits.

Finalmente, el tamaño d del campo desplazamiento se puede calcular de la siguiente forma:

$$d = n - s = 16 - 3 = 13 \text{ bits.}$$

Luego

| Nº de segmento | Desplazamiento |
|----------------|----------------|
| $s = 3$ bits | $d = 13$ bits |

- b) i) En primer lugar hay que comprobar que la dirección lógica $(0, 4526)$ es válida, para ello se compara el desplazamiento de esta dirección con la longitud del segmento 0 dada en la tabla de segmentos. Puesto que $4526 < 7230$ la dirección lógica es válida.

Finalmente, la dirección física se obtiene sumando la base del segmento 0 con el desplazamiento de la dirección lógica, es decir, $0 + 4526 = 4526$.

Luego la dirección lógica $(0, 4526)_{10}$ equivale a la dirección física **4526_{10}**

- ii) En primer lugar hay que comprobar que la dirección lógica $(3, 400)_{10}$ es válida, para ello se compara el desplazamiento de esta dirección con la longitud del segmento 3 dada en la tabla de segmentos. Puesto que $400 > 356$ se tiene un error de direccionamiento ya que se está violando el tamaño del segmento. Luego la dirección lógica $(3, 400)_{10}$ **no es válida**.