

**Evaluación NO continuada**

**EJERCICIO 4 (4 puntos)**

**Tiempo estimado: 90 m**

Tenemos los tres procesos representados en la figura. En el código de los procesos se especifica, para cada línea de código, el espacio de memoria que ocupa y el tiempo que empleará de CPU o E/S. En algunos casos se consideran despreciables (0 bytes, 0 ms.). El sistema reserva un espacio de 2k bytes para la pila de cada proceso. Las líneas indicadas como "codigo\_sin\_E/S" se considera que son un conjunto de instrucciones que no realizan ninguna operación de E/S que pueda bloquear al proceso. En el cronograma se muestra una posible ejecución de estos tres procesos, en recuadros se representan los tiempos de CPU y con líneas continuas los tiempos de E/S.

```

Proceso P1
-----
M: array [1050] of byte; /* 1050 bytes */
df,X: integer;          /* 4 bytes */

df = open("/Car/f1") /* 0 bytes;      CPU 0 ms. */
codigo_sin_E/S       /* 200k bytes; CPU 20 ms. */
for i = 1 to 4 do    /* 0 bytes;      CPU 0 ms. */
  read (df,X,2)     /* 0 bytes;      E/S 50 ms. */
  codigo_sin_E/S    /* 1k bytes;     CPU 5 ms. */
end for;            /* 0 bytes;     CPU 0 ms. */
close (df)          /* 0 bytes;     CPU 0 ms. */
    
```

```

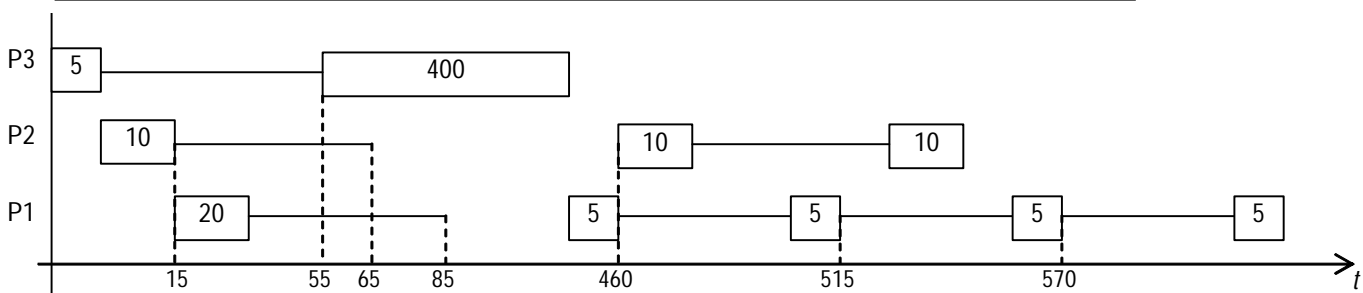
Proceso P2
-----
M: array [500] of byte; /* 500 bytes */
df,A: integer;         /* 4 bytes */

df = open("/Car/f2") /* 0 bytes;      CPU 0 ms. */
codigo_sin_E/S       /* 20k bytes;    CPU 10 ms. */
lseek (df,20050)     /* 0 bytes;     CPU 0 ms. */
for i = 1 to 2 do    /* 0 bytes;     CPU 0 ms. */
  read (df,A,2)     /* 0 bytes;     E/S 50 ms. */
  codigo_sin_E/S    /* 1k bytes;    CPU 10 ms. */
end for;            /* 0 bytes;     CPU 0 ms. */
close (df);         /* 0 bytes;     CPU 0 ms. */
    
```

```

Proceso P3
-----
M: array [500] of byte; /* 500 bytes */
df,H: integer;        /* 4 bytes */

df = open("/Car/f1") /* 0 bytes;      CPU 0 ms. */
codigo_sin_E/S       /* 1k byte;      CPU 5 ms. */
read (df,H,2)        /* 0 bytes;     E/S 50 ms. */
codigo_sin_E/S       /* 50k bytes;    CPU 400 ms. */
close (df);          /* 0 bytes;     CPU 0 ms. */
    
```



1.- Dado el cronograma de la figura, se pide:

- a) Indique una política de planificación de procesos que se ha podido seguir en el cronograma de la figura. Justifique la respuesta.
- b) Calcule el tiempo medio de espera, la eficiencia (utilización de la CPU) y el rendimiento (throughput).
- c) Supóngase que siempre existen en memoria un proceso P1, otro P2 y otro P3, en cuanto acaban de ejecutarse los tres, se crean otros tres nuevos. Calcule la utilización de la CPU basándose en el “modelo de multiprogramación” (utilice dos decimales). Compare el resultado con el cálculo de la eficiencia del apartado 1.a y explique brevemente a qué se debe la diferencia.

2.- Supongamos que el sistema utiliza una gestión de memoria con particiones variables con intercambio. Se dispone de una memoria de 2M bytes y la unidad de asignación es de 1k byte. En el instante cero se han creado alternativamente procesos iguales a P2 y P3 hasta completar la memoria.

- a) Queremos controlar las zonas de memoria que están libres y ocupadas por algún proceso. Indique dos técnicas que se pueden utilizar para gestionar la memoria libre/ocupada. Calcule, para cada una de ellas, la cantidad de memoria que necesitará para sus estructuras de datos en la situación descrita (se ha llenado la memoria de procesos igual a P2 y P3 alternados).
- b) Supongamos que un proceso es expulsado al área de *swap* en cuanto se bloquea en una E/S y un proceso que está en el área de *swap* solo se carga en memoria cuando está listo para ejecutarse y existe un hueco en memoria suficientemente grande para cargarlo. Si el proceso P1 llega justo en el instante en que la memoria se ha completado de procesos igual a P2 y P3, razone si podrá empezar a ejecutarse antes o después de los 400 ms. Se supone que la política de planificación de procesos que están en memoria es la misma que la utilizada en la figura.

3.- Supongamos que el fichero f1 ocupa 3 bloques de disco, el 9200, el 9201 y el 9209. El fichero f2 ocupa 22000 bytes para los que se han utilizado los bloques a partir del 3200 en adelante. El disco tiene un tamaño de 32G bytes dividido en bloques de 2k bytes. Para almacenar los i-nodos se utilizan los bloques desde 51 hasta 200, ambos inclusive. Para el directorio raíz se reserva el i-nodo 1. Para el resto de ficheros y directorios se utilizan los i-nodos a partir del 70.

- a) Supongamos que un i-nodo tiene los atributos del fichero, que ocupan 37 bytes, y además tiene 7 punteros directos, 1 indirecto simple y 1 indirecto doble. Indique el tamaño que deberá tener un i-nodo. Indique los valores que contendrá el i-nodo correspondiente al fichero f2.
- b) Indique cuáles son los bloques de disco que están siendo utilizados para almacenar la información correspondiente a los ficheros y directorios referenciados por los procesos P1, P2 y P3. Especifique el número de bloque (cuando se pueda conocer) y cuál es su contenido.
- b) Queremos comprobar la consistencia de bloques en el sistema de ficheros. Indique las estructuras de datos que serán necesarias y escriba el algoritmo (pseudocódigo) de una rutina que realice la comprobación.

4.- Para implementar el sistema de ficheros anterior, disponemos de un disco que tiene 5600 cilindros, 15 pistas por cilindro, 200 sectores por pista y 2k bytes por sector (cada cilindro contiene 3000 sectores). Como los bloques son de 2k bytes, cada bloque se almacena en un sector, de tal forma que el cilindro 0 contiene los bloques de 0 a 2999, el cilindro 1 contiene los bloques desde 3000 a 5999 etc.

- a) Indique, en orden cronológico, las cuatro primeras peticiones de lectura de disco que generarán entre los tres procesos P1, P2 y P3 con la instrucción ‘read’ durante la ejecución descrita en el cronograma (se supone que los tres procesos ya han ejecutado su llamada ‘open’ y P2 el ‘lseek’). Se deberá indicar el proceso que realiza la petición, el instante en que se hace la petición, el número de bloque accedido y el número de cilindro solicitado.
- b) La velocidad de rotación del disco es de 3000 rpm (revoluciones por minuto) y el tiempo de posicionamiento de 4 ms. por cada cilindro que se tenga que desplazar. Calcule el tiempo que empleará el disco en atender las dos primeras peticiones del apartado 4.a) para el caso de utilizar el algoritmo FCFS y para el caso de utilizar el algoritmo C-SCAN. Se supone que inicialmente la cabeza se encuentra en el cilindro 0.