

Se desea reutilizar el diseño de un sistema operativo de propósito general para llevar a cabo un sistema de tiempo real *soft* (es decir, un sistema en el que se deben intentar cumplir los plazos de las actividades que se definan y en el que la no consecución de los plazos degrada el sistema, pero no de manera crítica).

Supongamos que tenemos un sistema monoprocesador con el sistema operativo de tiempo real definido previamente. En dicho contexto tenemos una aplicación que se divide en las siguientes tareas que van a ejecutarse de manera concurrente (las tareas a su vez están divididas en actividades):

- Tarea A:
 - Abre un fichero *f1*, lo que implica una operación en disco [Actividad A1] (Plazo: 10 decimas de segundo)
 - Lee de una tubería *p1* inicialmente vacía [Actividad A2] (Plazo: 30 decimas de segundo)
 - Realiza una escritura de los datos leídos de la tubería en el fichero *f1*, lo que implica de nuevo una operación en disco [Actividad A3] (Plazo: 50 decimas de segundo)
 - Finalmente invoca la llamada `exit()` [Actividad A4] (No se define plazo para esta actividad)
- Tarea B:
 - Se trata de una tarea nueva, que debe ponerse en marcha [Actividad B1] (No se define plazo para esta actividad)
 - Una vez puesta en marcha la tarea, lee de una tubería *p2* vacía [Actividad B2] (Plazo: 20 decimas de segundo)
 - Abre el fichero *f1*, pero no necesita llevar a cabo ninguna operación en disco [Actividad B3] (Plazo: 40 decimas de segundo)
 - Lee bloque de fichero (requiere operación en disco) sobre variable global sin valor inicial, variable que nunca ha sido accedida y sólo ocupa una página. En el sistema hay marcos de páginas libres. El acceso a esa página implica un fallo que no requiere operación del disco [Actividad B4] (Plazo: 60 decimas de segundo)
 - La tarea continúa en modo usuario [Actividad B5] (No se define plazo para esta actividad)
 - Finalmente invoca la llamada `exit()` [Actividad B6] (No se define plazo para esta actividad)
- Tarea C:
 - Escribe en la tubería *p2* [Actividad C1] (Plazo: 10 decimas de segundo)
 - Escribe en la tubería *p1* [Actividad C2] (Plazo: 20 decimas de segundo)
 - Posteriormente sigue en modo usuario [Actividad C3] (No se define plazo para esta actividad)

Cada una de las actividades son planificables por el sistema operativo. Hay que tener en cuenta que las actividades correspondientes a cada una de las tareas tienen una relación de precedencia y que una actividad no se activa hasta que la anterior dentro de la tarea haya finalizado.

Las interrupciones de disco usan interrupción SW de sistema. Se utiliza un sistema con prioridades.

Responder **razonadamente** a las siguientes cuestiones:

1. (0.5) ¿Qué tipo de núcleo sería más adecuado para el sistema de tiempo real descrito?
2. (0.5) ¿Con qué abstracciones del sistema operativo se podrían implementar las tareas y las actividades asociadas a cada tarea en el sistema de tiempo real?
3. (2) Definir cómo se utilizarían las prioridades en el sistema de tiempo real. Indicar si se

- pueden dar problemas de inversión de prioridades y cómo podrían resolverse.
4. (1) ¿Qué aspectos habría que tener en cuenta para garantizar los plazos a nivel de tarea teniendo en cuenta que hay actividades que no requieren plazo y secuencialmente son anteriores a otras actividades con plazo dentro de dicha tarea?
 5. (1) ¿Qué diferencia habría si los plazos se definieran a nivel de tarea en lugar de actividad?
 6. (3) Realizar una traza temporal del procesador sin tener en cuenta la inversión de prioridades. En la traza especificar la tarea en ejecución, el modo en qué están dichas tareas (usuario o sistema), todos los tratamientos de eventos que ocurren durante la ejecución, la activación de interrupciones SW que sean necesarias, los cambios de contexto voluntarios (CCV) e involuntarios (CCI) que se producen y la actividad que se encuentra realizando en ese instante (AX, BX, ...). Para realizar la traza, considerar que la interrupción de disco llega cuando ha habido al menos dos cambios de modo en el procesador.
 7. (1) Indica cuáles serían las diferencias en la traza temporal del procesador si se tuviera en cuenta la inversión de prioridades.
 8. (1) ¿Para qué utilizan las interrupciones SW de sistema las interrupciones de disco?

SOLUCIÓN.-

1. El núcleo más apropiado sería un núcleo expulsivo, dado que tiene mejores tiempo de respuesta y latencia, tanto valores medio como máximo. Eso acota más el tiempo de espera y reduce la inversión de prioridades.
2. Las tareas se pueden implementar con la abstracción de proceso, dado que se trata de un conjunto de actividades relacionadas que comparten recursos, las cuales a su vez podrían implementarse como hilos o *threads* de núcleo.
3. Las prioridades deberían definirse de acuerdo a los plazos de las actividades, por lo que debería definirse a nivel de hilo de usuario. A menor plazo, mayor prioridad, para lograr cumplir dichos plazos. Podría darse inversión de prioridades, dado que una actividad de alta prioridad podría esperar por una actividad de baja prioridad (por ejemplo, si la primera actividad está leyendo de una tubería vacía y la segunda es la encargada de escribir en la misma). La manera de resolver la inversión de prioridades es a través de la herencia de prioridades, es decir, la prioridad de un proceso se debería calcular dinámicamente, asignándole como valor el máximo entre su prioridad estática y la prioridad de aquellos procesos que o bien posean algún recurso requerido por dicho proceso o que hagan alguna operación que desbloquee al proceso.
4. Las actividades para las que no se definen plazos, pero que son anteriores a otras actividades con plazo deberían heredar dicho plazo de la actividad inmediatamente posterior con plazo, de modo que se garantice en la mayor medida posible el cumplimiento de dichos plazos.
5. En el caso de que los plazos se definieran a nivel de tarea en lugar de actividad, la prioridad se definiría para los procesos, teniendo todos los hilos de usuario de un proceso la misma prioridad. La definición de prioridad a nivel de actividad (hilo), al ser de grano más fino, da mayor flexibilidad al sistema de tiempo real para planificar las actividades, lo que redundará en un mayor ajuste a las necesidades de las actividades del sistema, aunque podría implicar mayor número de cambios de contexto (al cambiar entre diferentes hilos de diferentes procesos).
6. Consideramos las siguientes prioridades (suponiendo un menor valor mayor prioridad) y que se planifica a nivel de hilo. A igualdad de prioridad entre dos hilos, se intenta no hacer cambio de proceso, para evitar la sobrecarga de un cambio de proceso.
P(A1)=1
P(C1)=1
P(B1)=2
P(B2)=2
P(C2)=2
P(A2)=3

$$P(B3)=4$$

$$P(A3)=5$$

$$P(B4)=6$$

$$P(A4)=7$$

$$P(B5)=7$$

$$P(B6)=7$$

$$P(C3)=7$$

Véase figura 1.

7. Tal y como están definidos los plazos de las actividades y por tanto las prioridades, no se produce ninguna inversión de prioridad, por lo que la traza sería exactamente igual.
8. Las interrupciones SW de sistema se utilizan para diferir la ejecución de operaciones no urgentes del tratamiento de interrupciones. En el escenario descrito, en el caso de las interrupciones de disco existirán determinadas actividades no urgentes que pueden diferirse.

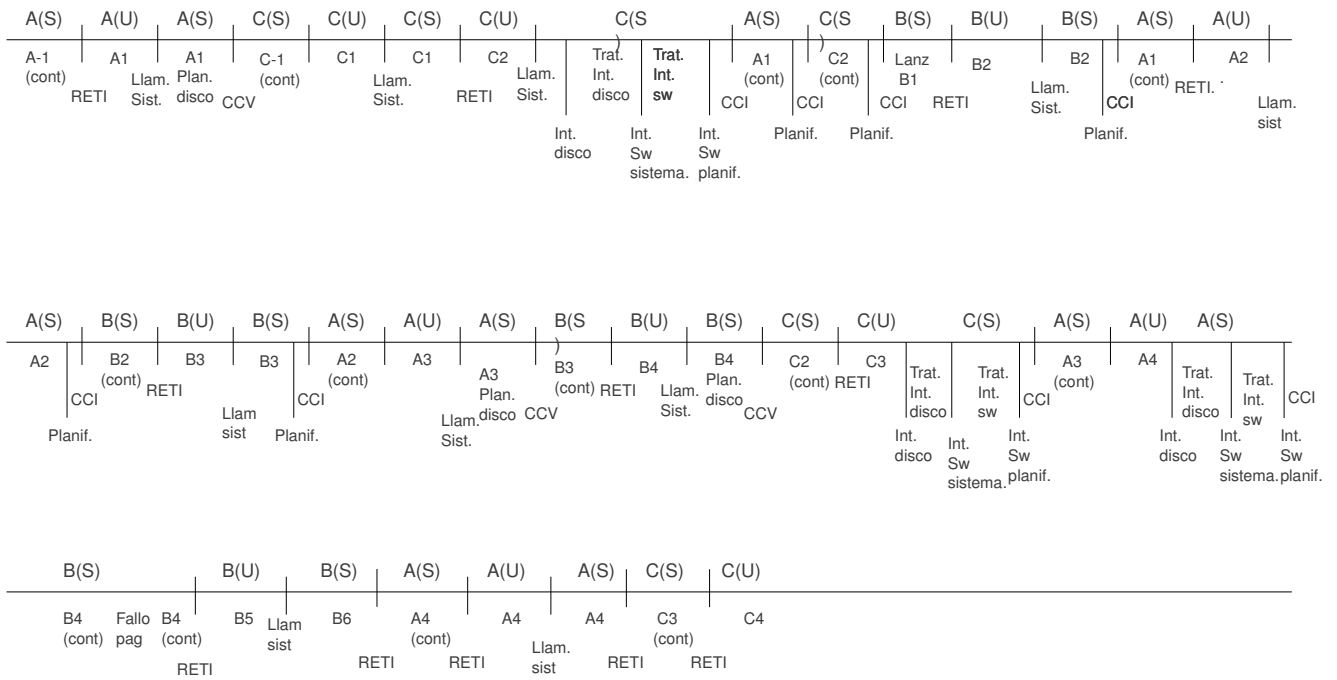


Figura 1: Traza temporal de ejecución