| Threads y semáforos | Máxima puntuación: 5 puntos |
|---|---|
| Descripción: | |
| Este proyecto consiste en implementar la interficie POSIX de threads y semáforos en ZeOS. Las llamadas al sistema que se tienen que implementar para threads son:<br>• Pthread_create (seguiremos la interficie POSIX, pero sin el parámetro pthread_attr_t *attr)<br>• Pthread_exit<br>• Pthread_join<br>Para los semáforos se tendrán que implementar:<br>• Sem_init<br>• Sem_wait<br>• Sem_post<br>• Sem_destroy<br><br>Para guardar los datos de management necesarios para los threads, se crearán estructuras TCB que se asignarán a los PCBs del proceso al cual pertenecen. Un proceso, como mucho, puede tener 10 threads y, en total, dentro del sistema, puede haber como mucho 20 threads. En este caso, el planificador será a dos niveles:<br>• El primer nivel seleccionará como siguiente thread a ejecutar uno dentro del mismo proceso. Cada thread tiene su propio quantum. Además, cada proceso tiene un quantum que indica cuanto tiempo seguido pueden estar, en total, todos sus threads utilizando la cpu. En el momento que el quantum por proceso expire, segundo nivel del planificador, se elegirá otro thread de otro proceso.<br><br>En cuanto a los semáforos, podrá haber, como máximo, 20 en el sistema. El sistema operativo, retornará un identificador de semáforo en la llamada sem_init que después se utilizará en sem_wait, sem_post y sem_destroy. | |
| 1er nivel (máximo 2 puntos) | |
| Diseña e implementa threads. Implementa juegos de prueba donde se vea la funcionalidad y el correcto funcionamiento de las llamadas al sistema para gestionar threads. | |
| 2º nivel (máximo 4 puntos) | |
| Diseña e implementa semáforos. Implementa juegos de prueba donde se vea la funcionalidad y el correcto funcionamiento de las llamadas al sistema para gestionar semáforos. | |
| 3er nivel (máximo 5 puntos) | |
| Diseña e implementa cargas de trabajo que te permitan calcular el tiempo de ejecución de las llamadas al sistema de creación y destrucción de threads. | |