

Primer Parcial Teórico

REPASO

Nota: El objetivo no es la pregunta misma, sino los conceptos que ella involucra

THREADS

1. Un thread no tiene program counter (PC) propio.
2. Un context switch entre threads, no requiere un context switch de registros
3. Un hilo creado por un proceso tendrá su propio contexto.
4. Un hilo creado por un proceso se ejecutará en el espacio de direcciones de este último.
5. Un hilo es la unidad básica de uso de la CPU.
6. Un hilo es la unidad de propiedad de recursos.
7. Dentro de un proceso, un hilo cuenta con un estado de ejecución.
8. Existe una PCB por proceso y por cada hilo que él cree.
9. Cuando un proceso se swapea, los hilos quedan en memoria en estado de espera.
10. En la administración de los hilos a nivel de usuario, interviene el kernel.
11. En los ULT, cada proceso se encarga de administrar sus hilos.
12. La suspensión de un ULT, provoca la suspensión del proceso.
13. En los KLT, el context switch entre hilos, no provoca un cambio de modo.

MULTIPROCESADORES

14. Los KLT, en un ambiente multiprocesador, pueden ejecutarse en distintos procesadores.
15. En multiprocesadores, en la organización maestro esclavo, una syscall puede ser atendida en cualquiera de los procesadores.
16. En multiprocesadores, si cada CPU tiene su SO es posible que una CPU este saturada y otras sin trabajo productivo.
17. En multiprocesadores, la técnica de SMP no requiere de exclusión mutua para el acceso a las estructuras del kernel.
18. No existen diferencias en la planificación de procesos entre SO monoprocesadores y multiprocesadores
19. En un sistema distribuido, todos los SO de las diferentes computadoras que participan deben ser iguales.

20. En las multicomputadoras, cada CPU tiene su memoria.
21. En multicomputadoras, la comunicación entre procesos se realiza por:
a) Memoria Compartida b) Pasajes de Mensajes c) RPC d) Ninguna
22. En multicomputadoras, cada nodo puede correr un SO diferente.
23. Las computadoras que forman una Grid deben ser todas iguales
24. El middleware es una capa de software entre el Hardware y el Sistema Operativo

DEADLOCKS

25. Basta que una de las 4 condiciones de deadlock se cumpla, para que haya deadlock.
26. La desventaja de usar algoritmos de prevención del deadlock, es que baja el grado de multiprogramación.
27. En un esquema de una instancia por tipo de recurso, cuando se encuentra un ciclo en un grafo de asignación de recursos, la asignación de los recursos solicitados:
a- puede poner al sistema en estado inseguro
b- pone al sistema en estado inseguro.
28. Todos los estados inseguros son deadlock.
29. El algoritmo del Banquero sirve para sistemas con múltiples instancias de cada recurso.
30. Siempre que el grafo de recursos tiene ciclos, hay deadlock.