

Test de Concurrencia de Procesos

Sistemas Operativos Enero 2022

INSTRUCCIONES PARA EL TEST

- Leer cuidadosamente las preguntas
- No prisa en responder: mirar la teoría o por Internet las posibles respuestas, **razonando** lógicamente la respuesta
- Responder las respuestas que se crean convenientes y poner la **solución en un papel** con el N° de la pregunta y el n° de la(s) respuesta(s) correspondiente(s)
- Las respuestas serán discutidas en Clase.

1. Un Semáforo es:

- 1) Un mecanismo de Sincronización de Procesos
- 2) Un mecanismo de Intercambio de información entre Procesos
- 3) Un mecanismo de prevención de interbloqueos
- 4) Un mecanismo de multiprogramación

2. Un semáforo se parece a un monitor en que:

- 1) Son dos mecanismos para asegurar la exclusión mutua.
- 2) Son dos mecanismos de planificación de E/S.
- 3) Son mecanismos de gestión de los procesos
- 4) Se usan para controlar la multiprogramación

3. El monitor respecto al semáforo es:

- 1) De espera no activa
- 2) De espera activa
- 3) Menos seguro
- 4) Más seguro

4. El mailbox o intercambio de mensajes se basa en :

- 1) Semáforos y memoria compartida
- 2) Planificación de Comunicaciones
- 3) Gestión de Entrada/Salida
- 4) Planificación a largo plazo

5. Los mensajes o mailbox son mecanismos de :

- 1) Cooperación entre Procesos
- 2) Sincronización entre Procesos
- 3) Exclusión Mutua entre Procesos
- 4) Multiprogramación

6. La filosofía del monitor es similar a:

- 1) La de los sistemas de monitorización del uso de la memoria
- 2) La de los programas que muestran el estado del Sistema
- 3) La de los Semáforos
- 4) La del Monitor Residente

7. La condición de “exclusión mutua” entre 2 Procesos, nos indica:

- 1) La imposibilidad de cargar sus programas en los mismos frames de memoria
- 2) La imposibilidad de que usen el mismo tipo de recurso (impresoras, discos, etc)
- 3) La necesidad de sincronizar su ejecución para enviarse datos
- 4) La imposibilidad de ejecutarlos a la vez

8. Una Región Crítica es:

- 1) La Zona de exclusión mutua
- 2) Un algoritmo de espera activa
- 3) Un mecanismo que ofrece el sistema operativo para sincronizar procesos
- 4) Las tres respuestas anteriores son falsas

9. Un monitor condicional es:

- 1) Una construcción del sistema operativo
- 2) Una facilidad que ofrece el lenguaje de programación
- 3) Las 2 anteriores son falsas
- 4) Las 2 primeras son verdaderas

10. La Concurrency es:

- 1) Varios programas que necesitan acceder al mismo recurso
- 2) Muchos programas ejecutándose en el sistema
- 3) Se deben dar las dos condiciones anteriores
- 4) Varios programas que se sincronizan entre sí

11. En un Sistema Multiprocesador:

- 1) Se pueden presentar problemas de concurrency
- 2) Siempre habrá problemas de concurrency
- 3) Nunca habrá problemas de concurrency
- 4) Las 3 anteriores son falsas

12. Los Monitores y las Regiones Críticas tienen como característica común:

- 1) Son mecanismos de sincronización de espera no activa
- 2) Tienen una variante basada en condiciones
- 3) La debe soportar el lenguaje de programación
- 4) Las 3 anteriores son verdaderas

13. El Algoritmo de la Alternancia:

- 1) Falla en la condición de tiempo
- 2) Falla en la condición de progreso
- 3) Aunque falla se puede usar sin problemas
- 4) No falla, es un algoritmo usado en algunos Sistemas

14.El Algoritmo de Dekker:

- 1) Es un algoritmo de espera no activa
- 2) Es un algoritmo que falla en la condición de progreso
- 3) Aunque falla se puede usar SIN PROBLEMAS
- 4) NO FALLA, es un algoritmo usado en algunos sistemas

15. El algoritmo con Mutex:

- 1) Falla en la condición de tiempo
- 2) Falla en la condición de progreso
- 3) Aunque falla se puede usar SIN PROBLEMAS
- 4) NO FALLA, es un algoritmo usado en algunos Sistemas

16. El algoritmo de Lamport:

- 1) Es de espera no activa
- 2) Sirve para sincronizar 2 procesos
- 3) Sirve como base de las Regiones Críticas
- 4) Sirve para sincronizar varios procesos

17.El algoritmo de Peterson:

- 1) Es de espera no activa
- 2) Sirve para sincronizar 2 procesos
- 3) Sirve como base de las Regiones Críticas
- 4) Sirve para sincronizar varios procesos

18.La instrucción test & set:

- 1) Permite sincronizar varios procesos
- 2) Es un algoritmo de espera activa
- 3) No se suele utilizar por su complejidad
- 4) Incumple la condición de tiempo por ser un algoritmo interrumpible

19.La instrucción Lock:

- 1) Es la base del algoritmo de Petterson
- 2) Se implementa como un algoritmo de espera activa
- 3) Sirve para sincronizar SÓLO 2 PROCESOS
- 4) Se basa en la instrucción test & set

20.Las llamadas remotas:

- 1) Permiten sincronizar 2 procesos por medio del paso de mensajes
- 2) Es un mecanismo de sincronización entre 2 procesos que se ejecutan en ordenadores diferentes
- 3) La información entre los procesos se pasa a través de mensajes y una vez recibidos, se ejecuta la acción remota
- 4) Las 3 anteriores son verdaderas

21.Al sincronizar procesos con un algoritmo de “espera no activo”, el estado del proceso en espera será:

- 1) Preparado
- 2) Bloqueado
- 3) Suspendido bloqueado
- 4) Suspendido preparado

22. Dos programas que se ejecutan simultáneamente en un sistema multiprocesador, deben acceder (leer y escribir) a los mismos datos en disco.

Para asegurar el funcionamiento correcto:

- 1) Deben usar el mismo Compilador
- 2) Deben asegurar la exclusión mutua
- 3) No es necesario tomar precaución alguna
- 4) Cada uno debe poder acceder a la memoria del otro

23. La “exclusión mutua” asegura en un Sistema:

- 1) Que 2 o más procesos no accederán a las mismas direcciones de memoria.
- 2) Que un proceso sólo se podrá ejecutar cuando finalice el que le excluye
- 3) Que 2 procesos no se interferirán durante su ejecución
- 4) Que un disco no realiza dos operaciones de E/S simultáneamente

24. La Concurrencia de Procesos exige:

1. Que los procesos usen programas distintos
2. Que los procesos no se interfieran entre sí
3. Interceptar las interrupciones de los periféricos
4. Que el Sistema disponga de varios procesadores

25. La Sección Crítica de un Proceso es:

- 1) La parte de su código que más veces se ejecuta
- 2) La parte del programa que contiene las variables del mismo
- 3) La parte del código del mismo que primero se carga en memoria
- 4) El del proceso que utiliza recursos compartidos con otros procesos.

26. Durante la ejecución de ciertas operaciones, el procesador NO puede ser interrumpido. Elige las que creas que no puede serlo:

- 1) Cambiar de contexto
- 2) Ejecutar una operación de E/S
- 3) Ejecutar una instrucción de Semáforos
- 4) Ejecutar una Sección Crítica

27. La característica de “espera activa” es típica de :

- 1) Las instrucciones WAIT y SIGNAL
- 2) Los Semáforos
- 3) El algoritmo de Dekker
- 4) Los monitores

28. Si existen Secciones Críticas Anidadas (es decir, un Proceso en su Sección Crítica, intenta acceder a otra Sección Crítica ¿Cuál de estos métodos de Sincronización NO funciona?:

- 1) Los monitores
- 2) Deshabilitar interrupciones
- 3) Algoritmo de Petterson
- 4) Los Semáforos

29. Los Semáforos Binarios permiten el acceso de “m” procesos a su Sección Crítica, siempre y cuando sólo “n” procesos necesiten compartir el recurso.

Elige los valores de “m” y “n” adecuados:

- 1) $m=1, n=2$
- 2) $m=1, n=1$
- 3) $m=2, n=2$
- 4) $m=2, n > 1$

30. Un Semáforo con instrucciones P y V atómicas:

- 1) No asegura la exclusión mutua
- 2) Se convierte en un algoritmo de espera activa
- 3) Es un monitor
- 4) Actúa correctamente

31. Los Semáforos son herramientas para la Sincronización de Procesos, y por tanto, el Sistema:

- 1) Mantiene Semáforos por cada proceso activo.
- 2) Mantiene Semáforos para los recursos no compartibles
- 3) Sólo existen en los Sistemas Multiprocesadores
- 4) Mantiene Semáforos para los procesos bloqueados

32. Cuando un Proceso ejecuta una instrucción de Semáforo:

- 1) Queda en estado preparado
- 2) Queda en estado bloqueado
- 3) Permanece en ejecución hasta que pueda acceder al recurso
- 4) Ninguna de las anteriores es válida

Fin del Test de Concurrency