

Guatemala, 25/02/2026

Sistemas Operativos, sección 10

Eliazar Canastuj 23384

Melisa Mendizabal 23778

Renato Rojas 23813

### **Respuestas a preguntas: Lab 2.2**

1. Identifique claramente la sección crítica en su código.

La sección crítica es la misma tanto para lectores como para escritores. La de escritores, es cuando los profesores entran a actualizar la nota del curso; no pueden haber dos escritores editando. La de lectores es cuando entran y leen dicha nota del curso, pero es más permisiva que la de escritores, pues pueden haber varios lectores leyendo al mismo tiempo el recurso.

- 2. ¿Cómo se implementa la exclusión mutua?

Por medio del algoritmo de Lamport Bakery junto con los mutex. De esta manera, se asegura que cada hilo tenga su turno para poderse ejecutar y sus semáforos, evitando así que existan dos procesos accediendo al mismo recurso

- 3. ¿Qué variable compartida requiere protección?

Las notas del portal, ya que es el recurso principal el cual se accede (escribe y lee) y debe de estar protegido con semáforos.

- 4. ¿Dónde podría ocurrir una condición de carrera?

Actualmente, en ninguno. Pero con una mala implementación: Al momento de escribir la nota; y al momento de escribir la cantidad/conteo de lectores y escritores.

- 5. ¿Existe posibilidad de deadlock? Explique.

En el actual, no puede existir deadlock. Esto porque no hay ningún recurso que no se libere en algún momento, pues, aun sea tardado, siempre se liberarán los recursos.

- 6. ¿Existe posibilidad de starvation? Justifique.

Sí, para los lectores. Ya que el programa es con prioridad para escritores, los lectores se pueden quedar eternamente esperando (starvation) recursos.

- 7. ¿Qué ocurriría si se duplican los hilos/procesos?

Se consumiría mucho más CPU, principalmente porque implementa busy waiting. No debería por qué dar un error, pero sería más cargado y tardado para la computadora.

- 8. ¿Cómo se comporta el sistema bajo alta carga?

Igual que la respuesta pasada, no presenta errores, pero es mucha más carga de trabajo para el procesador, a consecuencia del busy waiting.

- 9. Diferencie concurrencia y paralelismo en su implementación.

Al pasar los lectores por el momento crítico, existe concurrencia, pues solo puede acceder al recurso un único hilo a la vez, pero es tan rápido con tantos hilos, que pareciera que es paralelo.

Paralelismo ocurre al momento de la creación de los hilos, así como en cada hilo haciendo un spin.

- 10. ¿Qué métricas demuestran que su solución es correcta?

Las que se imprimen al final, que miran:

- Nunca más de 1 escritor simultáneo (max\_escritores\_simultaneos debería ser 1).
- Nunca lectores y escritores al mismo tiempo.
- Puede haber múltiples lectores simultáneamente, con conteo total de lecturas y máximo de lecturas al mismo tiempo.
- El sistema mantiene progreso (hay lecturas y escrituras).
- Si algo falla, hay mensajes de error para documentar en dónde falló.