

# Práctica 3

Laura Itzel Rodríguez Dimayuga

---

- Tú solución propuesta cumple:
  - Exclusión mutua
  - No deadlock
  - Libre de hambruna

De ser así demuestra cada propiedad.

**Solución.** Las propiedades se cumplen por el Lock que ya esta implementado en Java, aunque en la practica se puede saturar y a veces puede dejar de ser *Fair*

---

Además contesta las siguientes preguntas justificando lo siguiente

- ¿Tu solución cumple para n prisioneros?
- Si tu programa tarda mucho en terminar porque crees que pasa eso ?
- De la pregunta anterior, que podrías proponer para mejorar los tiempos.

Estas preguntas debes contestarlas para cada version del problema.

**Solución.**

- a) Si, aunque para muchos prisioneros podría romperse el candado y que haya dos personas que entren en la habitación al mismo tiempo.
  - b) Aunque en teoría el lock sea justo e intenta que casi todos los prisioneros pasan y no se bloquen, en la practica cada prisionero tiene  $1/n$  posibilidades de pasar, y como necesitamos que cada prisionero pase 2 veces, este proceso puede tardar.
  - c) Podríamos hacer que si un prisionero ya paso 2 veces deje de intentar entrar a la habitación pero esto modifacaría las condiciones del problema.
- 

Por último analiza bien el siguiente enunciado:

“Si los candamos cumplen con exclusión mutua, no deadlock o libre de hambruna, es decir, con las propiedades para un candado seguro, entonces el sistema donde lo utilizemos tambien las cumplira”

De esto justifica porque si se cumple o porque no. Añade lo aprendido en esta práctica, así como las dificultades que tuviste para realizarla.

**Solución.** Un sistema cumple con las propiedades de exclusión mutua, no deadlock y libre de hambruna solo si todas las secciones críticas concurrentes están protegidas por candados que cumplen con estas propiedades. Si hay otras secciones del código que son

concurrentes y no están protegidas por los mismos candados seguros, no podemos garantizar que todo el sistema cumpla con estas propiedades.