

1. Hacer un programa que corra un thread con el doble tamaño de stack que el default.
2. Diseñar e implementar una estructura y su api con semántica de semáforo utilizando los recursos mutex y condition variable de pthread. Hacer un programa de prueba.
3. Diseñar e implementar un mutex utilizando un semáforo.
4. Diseñar e implementar una estructura y su api con semántica de condition variable utilizando semáforos. Hacer un programa de prueba.
5. Qué pasa cuando se descomentan las líneas “return 0;” al final de la función main en los ejemplos tcreate.c y tdetached.c?
6. Que es lo que falla en race.c? Cómo se puede arreglar?

7. Implementar una estructura de cola circular con funciones get y put. Sincronizar los accesos con los mecanismos de pthread. Hacer un programa que arranque múltiples threads productores y consumidores de la cola.

<http://en.wikipedia.org/wiki/Read-copy-update> (RCU)

http://en.wikipedia.org/wiki/Non-blocking_synchronization