

CC4302 Sistemas Operativos – Tarea 2 – Semestre Otoño 2024 – Profs.: Mateu, Torrealba, Arenas

En una calle hay 10 estacionamientos contiguos al lado derecho (y ninguno al lado izquierdo). Se identifican como 0, 1, 2, 3, ..., 9. Los *pthread*s representan a los vehículos que solicitan estacionar llamando a la función *reservar*. Esta función recibe como parámetro el número de estacionamientos contiguos requeridos (*k*). Los vehículos son de distinto tamaño de modo que un *citycar* requiere solo un estacionamiento pero un camión podría requerir 6 estacionamientos contiguos. Si hay *k* estacionamientos contiguos disponibles, esta función los reserva y retorna de inmediato la identificación del primer estacionamiento en la serie otorgada (0, 1, 2, etc.). De lo contrario, *reservar* espera hasta que hayan *k* estacionamientos contiguos disponibles. Cuando un automovilista se va invoca la función *liberar* indicando el primer estacionamiento que ocupaba (*e*) y cuantos se le habían otorgado (*k*). Esta función libera todos los estacionamientos de la serie otorgada previamente a ese automovilista con *reservar*, y por lo tanto pueden ser otorgados a otros automovilistas.

Programa las funciones *reservar* y *liberar*. Sus encabezados son:

```
void initReservar(); //inicialización
void cleanReservar(); //limpieza
int  reservar(int k);
void liberar(int e, k);
```

Restricciones: Debe atender a los vehículos por orden de llegada. Además debe usar un mutex y **una sola condición**. No está permitido usar múltiples condiciones. Por lo tanto, los vehículos necesitarán sacar número para ser atendidos por orden de llegada (como en algunas farmacias).

Instrucciones

Baje *t2.zip* de U-cursos y descomprímalo. El directorio *T2* contiene los archivos *test-reservar.c*, *Makefile*, *reservar.h* (con los encabezados requeridos) y otros archivos. Ud. debe programar en el archivo *reservar.c* las funciones solicitadas. Defina otras funciones si las necesita.

Pruebe su tarea bajo Debian 12. Ejecute el comando *make* sin parámetros. Le mostrará las opciones que tiene para compilar su tarea. Estos son los requerimientos para aprobar su tarea:

- *make run* debe felicitarlo por aprobar este modo de ejecución.
- *make run-g* debe felicitarlo.

- *make run-san* debe felicitarlo y no reportar ningún incidente en el manejo de memoria.
- *make run-thr* debe felicitarlo y no reportar ningún datarace.

Cuando pruebe su tarea con *make run* en su computador asegúrese de que esté configurado en modo alto rendimiento y que no estén corriendo otros procesos intensivos en uso de CPU al mismo tiempo.

Invoque el comando *make zip* para ejecutar todos los tests y generar un archivo *equipo.zip* que contiene *equipo.c*, con su solución, y *resultados.txt*, con la salida de *make run*, *make run-g*, *make run-san* y *make run-thr*.

Entrega

Ud. solo debe entregar por medio de U-cursos el archivo *reservar.zip* generado por *make zip*. Recuerde descargar el archivo que subió, descargar nuevamente los archivos adjuntos y volver a probar la tarea tal cual como la subió a U-cursos. Solo así estará seguro de no haber entregado archivos incorrectos. Se descuenta medio punto por día de atraso. No se consideran los días de receso, sábados, domingos o festivos.