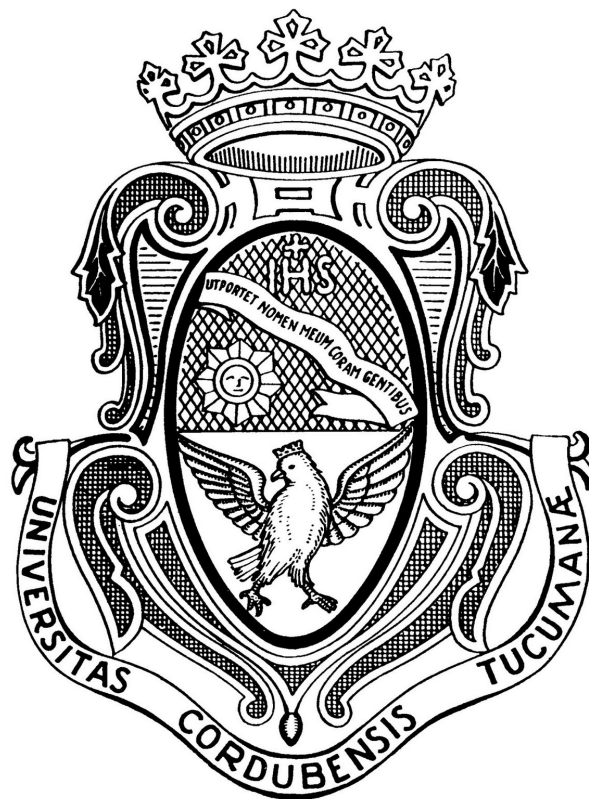


Universidad Nacional de Córdoba
Facultad de Ciencias Exactas, Físicas y Naturales



Trabajo Práctico Integrador

Programación Concurrente

Docentes:

Ing. Orlando Micolini

Ing. Luis Ventre

Alumnos:

Navarro, Sebastián

navarrosebastian95@gmail.com

Piñero, Tomás Santiago

tom-300@hotmail.com

Año 2019

Índice

Índice	1
1. Enunciado	2
2. Desarrollo	3
Tabla de estados.	3
Tabla de eventos.	4
Hilos	5
Red de Petri.	6
Políticas	6

1. Enunciado

En este práctico se debe resolver el control de acceso a una playa de estacionamiento con 3 entradas (calles) diferentes. En esta playa hay 2 pisos, y en cada piso pueden estacionar 30 autos. La playa cuenta con 2 salidas diferentes y una única estación de pago (caja). En los accesos a la playa y en los egresos existen barreras que deben modelarse.

La playa cuenta con lugares (3) donde los vehículos se detienen cuando quieren entrar (barrera), una vez que ingresaron se les indica un piso y estacionan (puede ser piso 1 o piso 2). Se debe cuidar que no se permita el ingreso (superar barrera) a más vehículos de los espacios disponibles totales.

Los autos que se retiran de la playa deben liberar un espacio del piso en que se encontraban (diferenciar estacionamiento en cada piso). Cuando un vehículo se va a retirar puede optar por salida a calle 1 o salida a calle 2. Luego debe abonar la estadía. El cobro de la estadía le lleva a un empleado promedio al menos 3 minutos. (Existe una sola caja)

En caso de que la playa esté llena, se debe encender un cartel luminoso externo que indica tal situación.

El sistema controlador debe estar conformado por distintos hilos, los cuales deben ser asignados a cada conjunto de responsabilidades afines en particular. Por ejemplo: Ingreso de vehículos, manejo de barreras, etcétera.

Debe realizar:

1. La red de Petri que modela el sistema.
2. Agregar las restricciones necesarias para evitar interbloqueos ni accesos cuando no hay lugar, mostrarlo con la herramienta elegida y justificarlo.
3. Simular la solución en un proyecto desarrollado con la herramienta adecuada (explique porque eligió la herramienta usada).
4. Colocar tiempo en las estación de pago caja (en la/s transición/es correspondiente/s).
5. Hacer la tabla de eventos.
6. Hacer la tabla de estados o actividades.
7. Determinar la cantidad de hilos necesarios (justificarlo)
8. Implementar dos casos de Políticas para:
 - Prioridad llenar de vehículos planta baja (piso 1) y luego habilitar el piso superior. Prioridad salida indistinta (caja).
 - Prioridad llenado indistinta. Prioridad salida a calle 2.
9. Hacer el diagrama de clases.
10. Hacer los diagramas de secuencias.
11. Hacer el código.
12. Hacer el testing.

2. Desarrollo

Tabla de estados.

Numero	Plaza	Estado
P0	Limitador clientes calle 1	Buffer que limita la cantidad de clientes a ingresar por la primer calle
P1	Limitador clientes calle 2	Buffer que limita la cantidad de clientes a ingresar por la segunda calle
P2	Limitador clientes calle 3	Buffer que limita la cantidad de clientes a ingresar por la tercer calle
P3	Autos esperando ingresar calle 1	Cola de autos esperando para sacar el ticket en la primer calle
P4	Autos esperando ingresar calle 2	Cola de autos esperando para sacar el ticket en la segunda calle
P5	Autos esperando ingresar calle 3	Cola de autos esperando para sacar el ticket en la tercer calle
P6	Auto pasando barrera 1	El cliente sacó el ticket y la barrera se levanta
P7	Auto pasando barrera 2	El cliente sacó el ticket y la barrera se levanta
P8	Auto pasando barrera 3	El cliente sacó el ticket y la barrera se levanta
P9	Barrera de calle 1	Mutex para el uso de la barrera de la primer calle
P10	Barrera de calle 2	Mutex para el uso de la barrera de la segunda calle
P11	Barrera de calle 3	Mutex para el uso de la barrera de la tercer calle
P12	Limitador de autos	Buffer que limita la cantidad de autos a utilizar la rampa
P13	Autos por tomar rampa	Autos utilizando la rampa
P14	Auto buscando lugar arriba	Cliente buscando lugar para estacionar el auto en planta alta
P15	Auto buscando lugar abajo	Cliente buscando lugar para estacionar el auto en subsuelo
P16	Rampa	Mutex de uso de la rampa
P17	Auto estacionado arriba	Auto estacionado en la planta de arriba
P18	Auto estacionado abajo	Auto estacionado en el subsuelo
P19	Lugares disponibles arriba	Cantidad de lugares disponibles en la planta alta
P20	Lugares disponibles abajo	Cantidad de lugares disponibles en el subsuelo
P21	Cliente pagando arriba	Cliente sale de la planta alta y abona el ticket
P22	Cliente pagando abajo	Cliente sale del subsuelo y abona el ticket
P23	Auto en rampa	Cliente esta bajando o subiendo por la rampa
P24	Cliente por pagar	Cliente esperando para pagar el ticket
P25	Cliente pagando	Cliente pagando el ticket

P26	Cajero	Cajero
P27	Auto por salir	Auto por abandonar la playa de estacionamiento
P28	Limitador de playa	Buffer que limita la cantidad de clientes que pueden ingresar al estacionamiento

Cuadro 1: Tabla de estados.

Tabla de eventos.

Numero	Plaza	Estado
T0	Entrar calle 1	Autos ingresando por la primer calle
T1	Entrar calle 2	Autos ingresando por la segunda calle
T2	Entrar calle 3	Autos ingresando por la tercer calle
T3	Sacar ticket 1	El cliente saca el ticket para levantar la barrera
v4	Sacar ticket 2	El cliente saca el ticket para levantar la barrera
T5	Sacar ticket 3 3	El cliente saca el ticket para levantar la barrera
T6	Levantar barrera 1	Una vez sacado el ticket, se levanta la barrera para el ingreso a la playa
T7	Levantar barrera 2	Una vez sacado el ticket, se levanta la barrera para el ingreso a la playa
T8	Levantar barrera 3	Una vez sacado el ticket, se levanta la barrera para el ingreso a la playa
T9	Subir por rampa	El cliente sube a la planta alta del estacionamiento
T10	Bajar por rampa	El cliente va a la planta baja del estacionamiento
T11	Buscar lugar arriba	El cliente busca lugar para estacionar en la planta alta
T12	Buscar lugar abajo	El cliente busca lugar para estacionar en la planta baja
T13	Aguardar salida arriba	El cliente dejó el auto estacionado en la planta alta
T14	Aguardar salida abajo	El cliente dejó el auto estacionado en la planta baja
T15	Bajar	El cliente de la planta alta se dirige a la salida
T16	Subir	El cliente de la planta baja se dirige a la salida
T17	Esperar caja	El cliente se dirige a la caja
T18	Pagar	El cliente paga el ticket en la caja
T19	Devolver cajero	El cliente elige la salida
T20	Salir por calle 1	El cliente sale por la calle 1
T21	Salir por calle 2	El cliente sale por la calle 2

Cuadro 2: Tabla de eventos.

Hilos

Los hilos son aquellos que realizan un conjunto de acciones, por lo que para determinar la cantidad de hilos necesarios, se realizó una modularización de la red en actividades, por lo tanto, la cantidad de hilos utilizados son 8

1. Entrada calle 1: T0 - T3.
2. Entrada calle 2: T1 - T4.
3. Entrada calle 3: T2 - T5.
4. Asignar piso: T6 - T7 - T8 - T9 - T10.
5. Planta Alta: T11 - T13 - T15.
6. Planta Baja: T12 - T14 - T16.
7. Cobrar tickets: T17 - T18 - T19.
8. Salida: T20 - T21.

Hilos 1, 2 y 3.

Los 3 hilos iniciales en las entradas son de existencia trivial, donde cada calle (o entrada) es independiente una de la otra, y cada una cuenta con sus propios recursos. La red resultante en este bloque será el estado inicial para el hilo 4.

Hilo 4.

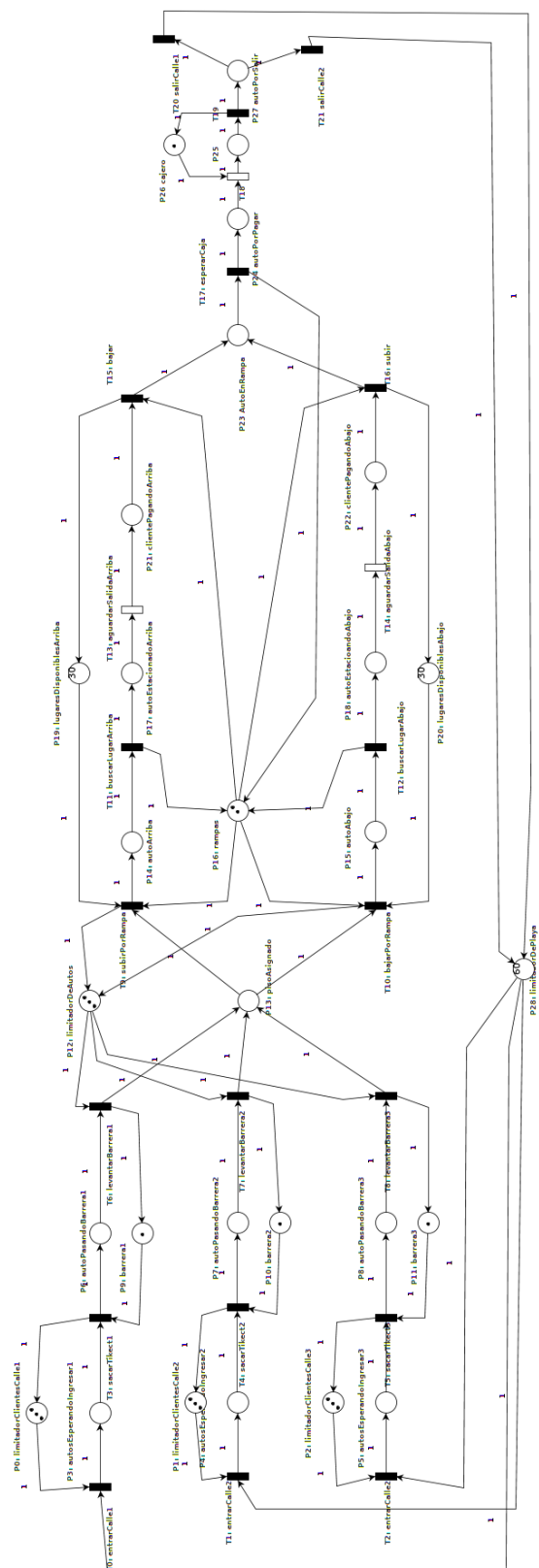
La segunda etapa de la red, el tramo que abarca el acceso y la asignación de piso, arriba o abajo, es ejecutada mediante un solo hilo. Este hilo resulta indispensable para la fluidez de la red. Al ser el conjunto de transiciones con más actividad, resulta necesario contar con un hilo que siempre tenga alguna de sus transiciones sensibilizadas, de modo que éste se encole la menor cantidad de veces posible y en consecuencia, los alrededores de la red no conduzcan a un interbloqueo. Además, al asignarle las transiciones T9 y T10 que introducen conflicto, facilita la implementación de políticas para resolverlo.

Hilos 5, 6 y 7.

Se independizan esos 3 conjuntos de transiciones de la red para modularizar lo más posible y así elevar la concurrencia de la red.

Hilo 8.

Se asigna un hilo a la dupla de transiciones T20 y T21 para resolver el último conflicto de la red, de modo que se puede implementar más fácilmente una política que las administre, y que además asegure un comportamiento consistente.



Haremos uso de una clase dedicada a cronometrar aquellas transiciones temporizadas.

Una transición con tiempo se cronometra desde el inicio de la sensibilización de la transición, y en el momento en el que se quiere disparar, se debe verificar que el tiempo dicho esté dentro del intervalo de tiempo $[a_i, b_i]$.

Si el disparo se quiere realizar antes del límite inferior (a_i), el hilo debe dormir $a_i - x$ segundos, donde x es el tiempo transcurrido desde que se sensibilizó la transición.

Semántica utilizada para las transiciones temporizadas.

Se tomo la primera de las semánticas especificadas en “*Redes_de_Petri_Temporales_2017.pdf*” - Hoja 11, por lo tanto el disparo se realiza en dos etapas:

- Transcurre un determinado tiempo desde que una transición se sensibiliza
- Se retiran y colocan las marcas de forma atómica

Cabe mencionar que esto se usa para ambos tipos de transiciones: inmediatas y temporales, ya que en el caso de la inmediata, el paso 1 implica un intervalo de tiempo nulo.

Políticas

Aquel conjunto de transiciones pertenecientes a un hilo en particular, que no requieran el tratado de políticas específicas, se dispararán de forma consecutiva siguiendo una secuencia de bucle para garantizar el orden y la fluidez de la red.

En el caso en que la prioridad de la transición sea indistinta, se elegirá la primera de las transiciones que corresponden al hilo segun el orden que llevan en el archivo ‘hilos.txt’, de modo que este orden puede suponerse como un orden de prioridades.

Por otro lado, las transiciones que requieren tratado de politicas especificas, son administradas por la clase Politicas del proyecto y obedecen al enunciado.