**Walkers of the city**

Programacion concurrente

**Integrantes:**

* Altman Quaranta Augusto.
* Fretes Ricardo.
* Diaz Marcos.

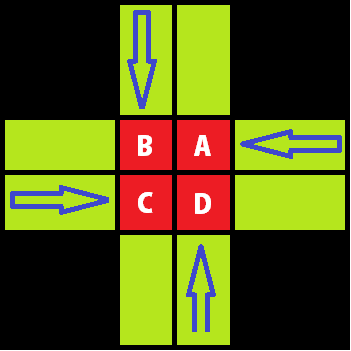
Córdoba, 23 de Noviembre del 2012

***Sistema***

**Introducción**

Se quiere representar una esquina (un cruce) transitada por autos que no deben chocar entra ellos. Los autos se comportan independientemente uno de otro, viajando a diferentes velocidades y realizando diferentes acciones sobre el cruce, como ser seguir derecho, doblar a la izquierda, derecha, etc.

Se plantea entonces un sistema como solución a lo planteado, en donde cada auto es un hilo de ejecución diferente, que viajan a velocidades aleatorias y ejecutan acciones aleatorias sobre el cruce. Este último además es un monitor que gestiona 4 recursos que componen la esquina, posibilitando la exclusión mutua entre los hilos y evitando los interbloqueos y la inanición.

Definimos el cruce como sigue:

En donde los cuadrados rojos A, B, C y D son coordenadas específicas en el cruce y son recursos que comparten todos los autos que deben ser gestionados por el monitor para obtener exclusión mutua. Se desea, para aumentar el nivel de paralelismo que los autos, vayan liberando recursos a medida que los utilizan y no los devuelvan todos juntos al final de la ejecución de la acción que estén llevando a cabo.

Los autos llegarán por cualquiera de las direcciones marcadas con flechas azules, y dependiendo de la misma van a entrar al cruce por un recurso diferente. Al entrar al cruce los autos pueden ejecutar una de las siguientes acciones:

* Seguir derecho
* Doblar a la derecha
* Doblar a la izquierda

A su vez dependiendo por donde entren los autos y que acción ejecuten van a tomar diferentes recursos formando diferentes secuencias de uso y liberación de los mismos.

En la siguiente tabla se resumen las secuencias de uso y liberación de recursos según el recurso de entrada y la acción a ejecutar.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Doblar a la derecha | Seguir derecho | Doblar a la izquierda |
| A | A | A, B | A, B, C |
| B | B | B, C | B, C, D |
| C | C | C, D | C, D, A |
| D | D | D, A | D, A, B |

Para evitar interbloqueos el monitor deberá asignarle todos los recursos que el auto requiera al mismo tiempo antes de comenzar a ejecutar la acción y los mismos deberán ser liberados uno a uno a medida que se van utilizando, por esto último y como se requirió más arriba se aumenta el nivel de paralelismo del sistema. La inanición se evitara aplicando una política de gestión de recursos que se explicara mas adelante.