

Práctica 2 - Puente de Ambite - Rodrigo de la Nuez Morelada

- El pueblo de Ambite tiene un puente que atraviesa el río Tajuña. Es un puente compartido por peatones y vehículos. La anchura del puente no permite el paso de vehículos en ambos sentidos. Por motivos de seguridad, los peatones y los vehículos no pueden compartir el puente. En el caso de los peatones, si que pueden pasar peatones en sentido contrario.
- Desarrolla en papel el diseño y el monitor (o monitores) necesarios. Parte de una solución sencilla que cumple la seguridad y, a partir de ella, intenta buscar soluciones a los problemas de iniciación.
 - Escribe el siguiente del monitor.
 - Demuestra que el puente es seguro (no hay coches y peatones a la vez en el puente, no hay coches en sentidos opuestos)
 - Demuestra la ausencia de deadlocks.
 - Demuestra la ausencia de iniciación.
- Implementa una solución en python con la biblioteca multiprocessing. → initial-bridge.py → final-bridge.py

monitor initial-bridge

```
{ number_north: int=0
  number_south: int=0
  number_pedestrian: int=0 }
```

```
{ condition_north: vc=True
  condition_south: vc=True
  condition_pedestrian: vc=True }
```

```
INV = {number_north >= 0 ^ number_south >= 0 ^ number_pedestrian >= 0 ^  

       number_north > 0 → (number_south == 0 ^ number_pedestrian == 0) ^  

       number_south > 0 → (number_north == 0 ^ number_pedestrian == 0) ^  

       number_pedestrian > 0 → (number_north == 0 ^ number_south == 0)}
```

- wants_enter_car(direction):

```
{INV  

if direction == north:  

  condition_north.wait(number_south == 0 ^ number_pedestrian == 0)  

  number_north += 1  

{INV ^ number_north > 0}  

else:  

  condition_south.wait(number_north == 0 ^ number_pedestrian == 0)  

  number_south += 1  

{INV ^ number_south > 0}  

{INV ^ (number_north > 0 ^ number_south > 0)}
```

- wants-enter-pedestrian():

```
{INV}
condition-pedestrian.wait(number-north == 0 & number-south == 0)
number-pedestrian += 1
{INV & number-pedestrian > 0}
```

- leaves-car(direction):

```
{INV & (number-north > 0 || number-south > 0)}
```

if direction == north:

```
{INV & number-north > 0}
number-north -= 1
condition-south.signal()
{INV}
```

else:

```
{INV & number-south > 0}
number-south -= 1
condition-north.signal()
{INV}
```

condition-pedestrian.signal()

```
{INV}
```

- leaves-pedestrian():

```
{INV & number-pedestrian > 0}
number-pedestrian -= 1
condition-north.signal()
condition-south.signal()
{INV}
```

Esta versión inicial cumple el requisito de seguridad pues para que entre un coche se obliga a que no haya peatones ni coches en dirección contraria en el puente. De igual manera, si un peatón quiere entrar no puede haber coches.

Para que hubiera algún deadlock todos los procesos tendrían que estar esperando =
= number-north, number-south, number-pedestrian > 0, por lo que entró
al menos uno de cada tipo sin salir = uno de ellos entró sin que su
condición fuera cierta (contradicción)

Podría haber problemas de inanición si, por ejemplo, los coches de norte y sur se
ponen de acuerdo para que no entran peatones al puente. Resolveremos este
problema atendiendo al número de peatones y vehículos esperando a entrar.

Para la versión definitiva mantendremos las variables del caso sencillo, y las funciones leaves-car(direction) y leaves-pedestrian() serán las mismas. Anadimos las siguientes variables al monitor definitivo:

monitor final_bridge

$$\left\{ \begin{array}{l} \text{number_north: int=0} \\ \text{number_south: int=0} \\ \text{number_pedestrian: int=0} \end{array} \right.$$

$$\left\{ \begin{array}{l} \text{north_waiting: int=0} \\ \text{south_waiting: int=0} \\ \text{pedestrian_waiting: int=0} \end{array} \right.$$

$$\left\{ \begin{array}{l} \text{condition_north: vc=True} \\ \text{condition_south: vc=True} \\ \text{condition_pedestrian: vc=True} \end{array} \right.$$

$$\left\{ \begin{array}{l} \text{nccondition: vc=True} \\ \text{sccondition: vc=True} \\ \text{pccondition: vc=True} \end{array} \right.$$

$$\text{INV} = \{ \text{number_north} \geq 0 \wedge \text{number_south} \geq 0 \wedge \text{number_pedestrian} \geq 0 \wedge \\ \text{north_waiting} \geq 0 \wedge \text{south_waiting} \geq 0 \wedge \text{pedestrian_waiting} \geq 0 \wedge \\ \text{number_north} > 0 \rightarrow (\text{number_south} = 0 \wedge \text{number_pedestrian} = 0) \wedge \\ \text{number_south} > 0 \rightarrow (\text{number_north} = 0 \wedge \text{number_pedestrian} = 0) \wedge \\ \text{number_pedestrian} > 0 \rightarrow (\text{number_north} = 0 \wedge \text{number_south} = 0) \}$$

- wants-enter-car(direction):

{INV4

if direction == north :

nccondition.wait (south_waiting == 0 \wedge pedestrian_waiting == 0)

north_waiting += 1

condition_north.wait (number_south == 0 \wedge number_pedestrian == 0)

north_waiting -= 1

number_north += 1

sccondition.signal()

{INV \wedge number_north > 0}

else :

sccondition.wait (north_waiting == 0 \wedge pedestrian_waiting == 0)

south_waiting += 1

condition_south.wait (number_north == 0 \wedge number_pedestrian == 0)

south_waiting -= 1

number_south += 1

nccondition.signal()

{INV \wedge number_south > 0}

pccondition.signal()

{INV \wedge (number_north > 0 \vee number_south > 0)}

- waits-exit-pedestrian():

{INV4}

pedestrian.wait (north-waiting == 0 \wedge south-condition == 0)
 pedestrian-waiting += 1
 condition-pedestrian.wait number-north == 0 \wedge number-south == 0)
 pedestrian-waiting -= 1
 number-pedestrian += 1
 condition.signal()
 scondition.signal()
{INV \wedge number-pedestrian > 0}

- Las funciones 'leaves-car(direction)' y 'leaves-pedestrian()' son como en 'initial-bridge'.

Al igual que en la versión sencilla, para que entre un coche se obliga a que no haya peatones ni coches en dirección contraria y para que pase un peatón no puede haber coches cruzando el puente. De esta manera, se cumplen las condiciones de seguridad.

No puede haber deadlocks según el número de coches en el puente siguiendo un mecanismo idéntico al caso análogo. De forma similar, si north-waiting, south-waiting, pedestrian-waiting > 0, simultáneamente, entonces alguno de ellos no habrá cumplido su correspondiente condición de espera contradicción.

Al considerar el número de vehículos y peatones esperando a entrar entramos que haya problemas de iniciación pues se le dará prioridad a los vehículos y peatones en espera cuando el puente quede vacío al no poder entrar otros coches del tipo actual, dirigidos así al cambio. Así, entramos que将会 haber procesos que ocuparán el puente o esperar indefinidamente hasta que los demás terminen.