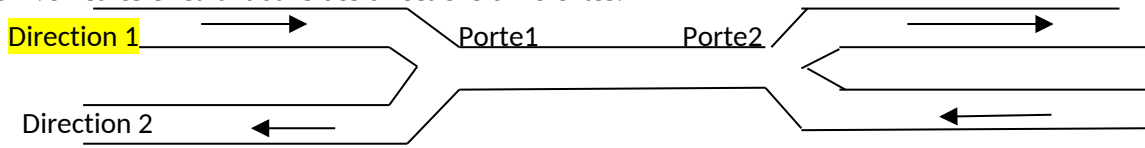


Exercice de TD (Brouillon de travail en ligne)

Exercice 1:

Nous considérons un pont de circulation à une seule voie sur lequel, il n'est pas possible d'autoriser le passage de deux véhicules circulant dans des directions différentes.



Direction 1 - -v5 ---V3--v2 >> -----à
 -----v0-----
 Direction2 <----- <<v1-----

Nous représentons les véhicules qui doivent emprunter ce pont par les processus suivants:

Processus Direction1	Processus Direction2
...
AccèsPont.Porte1() // ?est ce il n y a un vehicule de direction 2 sur le pont	AccèsPont.Porte2()
<circuler sur le pont>	<circuler sur le pont>
SortiePont.porte2	SortiePont.porte1()
....

1- Je suppose, dans un premier temps, uniquement 02 vehicule v0 et v1

D : semaphore binaire ;
 Init (D,1)

Processus v0 // sur direction1	Processus v1 // direction2
...
P (D) ; //Demander une autorisation d'AccèsPont	P (D) ; //Demander une autorisation
<circuler sur le pont>	<circuler sur le pont>
V(D) //SortiePont.porte2	V(D) ; //SortiePont.porte
....

Temps ↓	D=1 v0	D2 v1
		P(D), D=0
	P(D) ,D=-1	Traverse le pont
	vo dans <u>FA(D)</u>	Traverse le pont
		Traverse le pont
		V(D) ; D=0 et reveiller v0
	Traverser pont	
	Traverser pont ...	
	V(D) D=1 Aucun a reveiller	
	

2 – On suppose, **maintenant**, plusieurs vi dans les 2 direction

<pre> N1=0 ; D : semaphora binaire ; init(D,1) ; init(mutex1,1) ; Processus Vi-D1 // sur direction1, i :1....m ... N1++ ; Si N1=1 alors P (D) ; //Demander autorisation et bloque D2 Fsi ; <circuler sur le pont> N1-- ; Si N1==0 alors V(D) //debloque qcces D2 fsi </pre>	<pre> N2=0 ;; Processus V-D2 // direction2 N2++ Si N2=1 alors P (D) ; //Demander autorisation Bloque D1 <circuler sur le pont> N2-- Si N2==0 alors V(D) ; //debloquer D2 fsi </pre>
--	---

Déroulement :

----- à temps

D1

V0-D1 : N1= 1, P(D) D=0 traverser..... N1--= 2 fin-----
V1-D1 : N1=2, traverser.....N1--=1 fin----
V2-D1 : N1=3, traverser.....N1--=0, V(D) D++=0 reveiller V0-D2---

D2 :

V0-D2 : N2=1, P(D) D=-1 bloquétraverser...

**Problème N1=0

-----à temps -----à

V0-D1 : **N1=N1+1**(recupere val N1) et **ajouter 1 a N1**)**P(D)D=0** et il traverse

t1;N1=0 t3=N1+1=1

V1-D1 : **N1=N1+1**(recupere val N1) et **ajouter 1 a N1**) **P(D), D=-1èse bloquer !!!!**

t2 N1=0 t4 : N1+1=1

Problème : N1 est aussi une ressource (variable) partagée qu'il faut utiliser en exclusion mutuelle pour ne pas avoir des excusions incohérentes.

Solution :

<pre> N1=0 ; D, mutex1 : semaphora binaire ; init(D,1) ; init(mutex1,1) ; Processus Vi-D1 // sur direction1, i :1....m ... P(mutex1) ; //demande autorisation utilisation de N1 N1++ ; Si N1=1 alors P (D) ; //Demander autorisation et bloque D2 Fsi ; V(mutex1) ; //liberer utilisation de N1 <circuler sur le pont> P(mutex1) ; N1-- ; Si N1==0 alors V(D) //debloque qcces D2 fsi V(mutex1) ; </pre>	<pre> N2=0 ; mutex2 : sem binaire, init(mutex2,1) ; Processus V-D2 // direction2 P(mutex2) ; N2++ Si N2=1 alors P (D) ; //Demander autorisation Bloque D1 V(mutex2) ; <circuler sur le pont> P(mutex2) ; N2-- Si N2==0 alors V(D) ; //debloquer D2 fsi . V(mutex2) ;.... </pre>
---	---

Déroulement :

----à temps -----à

V0-D1 : P(mutex) mutex=0 **N1=N1+1** (recupere val N1) et ajouter 1 a N1) **P(D) D=0** V(mutex) reveiller V1 et il traverse

t1; N1=0

t3=N1+1=1

V1-D1 : P(mutex) mutex=-1 **bloque**.....

Debloque

N1=N1+1 (recupere val N1) et ajouter 1 a N1) P(D), D=-1 èse bloquer !!!!

t2 N1=0

t4 : N1+1=1

3 – On suppose, **maintenant**, plusieurs vi dans les 2 directions mais on rajoute l'hypothèse que le pont ne peut supporter plus de K véhicules a la fois.

Proposer une solution