

Corretion TD SEMAPHORE

EX1

Contexte commun

sémaphore mutex = 1, s= 0;

entier NbArrivés = 0; */*nbre de processus arrivés au rendez-vous*/*

Procédure RDV

Début

down(mutex);

NbArrivés = NbArrivés + 1;

Si (NbArrivés < N) Alors */* non tous arrivés */*

up(mutex); */* on libère mutex et */*

down(s); */* on se bloque */*

Sinon

up(mutex); */* le dernier arrivé libère mutex et */*

Pour i = 1 à N-1 Faire

up(s); */* réveille les N-1 bloqués */*

Finsi

Fin

EX2

void **SortirAvions ()** (

{

Down(Decollage_vide);

Ajouter_un_avion_dans_la_zone_attente_decollage();

Up(Decollage_plein);

}

void **Decollage ()**

{

Down(Decollage_plein);

Down(Piste);

Faire_decoller_un_avion ();

Up(Piste);

Up(Decollage_vide);

}

void **Atterrissage ()**

{

Down(Atterrissage_plein);

Down(Piste);

Faire_Atterrir_un_avion ();

Up(Piste);

Up(Atterrissage_vide);

}

void **AmenerAvions ()**

{

Down(Atterrissage_vide);

Ajouter_un_avion_dans_la_zone_attente_atterrissage();

Up(Atterrissage_plein);

}

EX3

Sémaphore SRC=1, SBO=1, SAS1=0, SAS2=0, libre=N, occupe=0, mutex=1;

<pre>mRC() { while (1) { down(SRC) ; RC() ; up(SAS1) ; } }</pre>	<pre>mBO() { while (1) { down (SBO) ; BO() ; up (SAS2) ; } }</pre>	<pre>mAS() { while (1) { down (SAS1) ; down (SAS2) ; P(libre) ; AS() ; up (SRC) ; up (SBO) ; up (occupe) ; } }</pre>	<pre>mEM() { while (1) { down (occupe) ; down (mutex) ; EM() ; up (mutex) ; up (libre) ; } }</pre>
--	--	--	---

EX4 :

Sémaphore autorisation =1 ;// Le sémaphore autorisation est partagé par tous les trains.

Sémaphore mutex =1 ;//Le sémaphore mutex est partagé par tous les trains allant dans le même sens.

// Il y a donc deux sémaphores mutex un pour chaque sens.

Demande d'accès par un train AversB

```
down(mutex)
Si NbAB =0 alors
down (autorisation)
F.SI
NbAB=NbAB+1 ;
up(mutex) ;
```

Sortie de la voie par B

```
down (mutex)
Si NbAB =1 alors
up (autorisation)
F.SI
NbAB=NbAB-1 ;
up(mutex) ;
```

Demande d'accès par un train BversA

```
down (mutex)
Si NbBA =0 alors
down(autorisation)
F.SI
NbBA=NbBA+1 ;
up(mutex) ;
```

Sortie de la voie par A

```
down (mutex)
Si NbBA =1 alors
up (autorisation)
F.SI
```

```
NbBA=NbBA-1 ;  
up(mutex) ;
```