

Série de TD N° 3

Exercice 1

On veut faire la gestion d'un parking de voitures contenant N emplacement. Chaque voiture est représenté par un processus. Synchronisez les différent processus en utilisant un moniteur.

Exercice 2

Réalisez un rendez-vous entre N processus en utilisant les moniteurs.

Exercice 3

Dans un bureau de poste un seul guichet seulement est réservé au versement des mandat et à l'achat des timbres. Le responsable du guichet vas servir les clients en alternance. On veut assimiler les clients à des processus parallèles et les synchroniser en utilisant les moniteurs . Écrire l'algorithmme de chaque client ainsi que l'algorithmme du moniteur.

Exercice 4 (Exercice tiré de l'examen Système D'exploitation II 2012-2013 de l'université de chlef)

Deux villes A et B sont reliées par une seule voie de chemin de fer.



Les règles de circulation sont les suivantes :

- La voie ne doit jamais être empruntée simultanément par deux trains allant en sens inverse,
- La voie peut être empruntée par un ou plusieurs trains allant tous dans le même sens,
- La priorité de parcours est la même pour les deux sens.

On considère deux classes de processus : les trains allant de A vers B : « **T-AB** » et les trains allant de B vers A : « **TB-A** ».

Processus T-AB	Processus T-BA
Début	Début
<code>Entree_A();</code> <code><Circulation sur la voie de A vers B></code> <code>Sortie_B();</code>	<code>Entree_B();</code> <code><Circulation sur la voie de B vers A></code> <code>Sortie_A();</code>
Fin.	Fin.

Q1) Quelle est la différence entre ce problème et le modèle des lecteurs/rédacteurs ?

Q2) En utilisant les sémaphores, écrire les codes des quatre procédures `entree_A()`, `entree_B()`, `sortie_A()` et `sortie_B()` de façon à ce que les processus respectent les règles de circulation sur la voie. Précisez clairement vos déclarations et initialisations.

Q3) Expliquer pourquoi la solution suivante (avec moniteurs) n'est pas correcte.

Moniteur AB ; Int nbA=0, nbB=0 ; Condition ca, cb ;	
Entree_A() { nbA++ ; si (nbB>0) alors ca.wait() fsi }	Entree_B() { nbB++ ; si (nbA>0) alors cb.wait() fsi }
Sortie_B() { nbA-- ; si (nbA==0) alors cb.signal() fsi }	Sortie_A() { nbB -- ; si (nbB==0) alors ca.signal() fsi }

Q4) Donnez une correction de la solution erronée.