Université de Chlef
Département Informatique
Filière : 3ème Année Licence TSW

Février 2013

Examen semestriel

Module de « Systèmes d'exploitation2 »

Corrigé

Exercice 1 : (12 points)

On considère le problème de synchronisation "Producteur/Consommateur" avec buffer limité, mais le nombre de producteurs est égal à 2 alors qu'il n'ya qu'un seul consommateur. Proposez une solution à ce problème en utilisant les sémaphores (Soignez vos déclarations et votre code).

On utilise pour ce problème 4 sémaphores :

Réponse :
Déclaration des variables utilisées :

- **Empty** : sémaphore (init à N , représentant le nombre de case vides du buffer)
- **Full** : sémaphore (init à 0, représentant le nombre de cases pleines du buffer).
- **MutexProd** : sémaphore d'exclusion mutuelle (init 1) pour protéger la variable In partagée par les producteurs.
- **In** : entier (init 0, représentant l'indice de l'élément pouvant recevoir un élément déposé par un producteur).
- **Out** : entier (init 0, représentant l'indice de l'élément prêt à être prélevé par un consommateur).

```
Processus Producteur i
Début
        Cycle
            Produire un message dans ZoneP
            Wait(Empty)
            Wait(MutexProd) ;
            Buffer[In] :=ZoneP ;
            In :=In+1 mod N ;
            Signal(MutexProd) ;
            Signal(Full)
        Fin Cycle
Fin.
```

```
Processus Consommateur
Début
        Cycle
                Wait(Full)
                ZoneC :=Buffer[Out] ;
                Out :=Out+1 mod N ;
                Signal(Empty) ;
                Consommer le message de ZoneC
        Fin Cycle
Fin.
```

Exercice 2 : ***(03 points)***

Quelles sont les critiques qu'on peut faire aux solutions logicielles et aux solutions matérielles du problème de l'exclusion mutuelle ? .

Réponse :

Critique des solutions logicielles : L'attente active : des cycles processeurs sont consommés inutilement pendant l'attente.

(1.5 points)

Critique des solutions matérielles : Les instructions matérielles sont spécifiques au matériel et ne peuvent pas être généralisées.

(1.5 points)

Exercice 3 : ***(05 points)***

Décrivez en quelques lignes comment peut-on réaliser les sémaphores en langage Java. Donnez le code en Java.

Réponse :

Les sémaphores n'existent pas en Java; il faut les déclarer à travers une classe. La classe contient l'attribut value et les 2 méthodes wait et signal. Wait et signal sont à utilisée avec l'option "synchronized" qui garantit l'accès en exclusion mutuelle à chaque méthode.

(2 points)

```
/* La classe Semaphore. */

public class Semaphore {
  private int value;

  public Semaphore(int n) {
    value = n;
  }

  public synchronized void wait() {
    while (value <= 0) {
      try { wait(); } catch (InterruptedException e) {};
    };
    value--;
  }

  public synchronized void signal() {
    value++;
    notifyAll();
  }
}
```

(3 points)