

Exercices classiques sur les sémaphores

Novembre 2008

1 Producteur & consommateur

Deux processus partagent une mémoire tampon circulaire de taille fixe. Le producteur produit des objets qu'il place dans le tampon n'est pas plein. Le consommateur retire des objets quand le tampon n'est pas vide.

- (a) Décrire un jeu de sémaphores pour gérer cette situation.
- (b) Ecrire un algorithme avec les fonctions P et V de Dijkstra.
- (c) Ecrire les codes des processus en langage C sous linux.

2 Feux de circulations

Un système de feux bicolores (rouge, vert) règle la circulation d'un carrefour à deux voies perpendiculaires en changeant de couleurs toute les 180 secondes. Une voiture peut s'engager dans le carrefour quand le feu est vert et si le carrefour est vide.

- Ecrire un algorithme fondé sur trois processus correspondant à un changement de couleur des feux, la traversée d'une voie et la traversée de l'autre voie. Ecrire les codes des processus en langage C sous linux.

3 Lecteurs & rédacteurs

On suppose une base de données utilisée par des lecteurs et des rédacteurs. Les lecteurs et les rédacteurs de cette base de données satisfont aux contraintes suivantes :

- plusieurs lecteurs doivent pouvoir lire la base de données en même temps ;
- si un rédacteur est en train de modifier la base de données, aucun autre utilisateur (ni rédacteur, ni mme lecteur) ne doit pouvoir y accéder.

Donner une solution qui donne la priorité aux lecteurs et s'appuie sur trois sémaphores et une variable, à savoir :

- Un sémaphore M_{Lect} , initialisé à 1 qui permet de protéger la variable $Lect$. Il s'agit donc d'un mutex.
- Un sémaphore M_{Red} , initialisé à 1 qui permet de bloquer les tches de rédaction. Il s'agit donc à nouveau d'un mutex.
- Un sémaphore Red , initialisé à 1 qui permet aussi de bloquer les tches de rédaction.
- Une variable $Lect$ qui compte le nombre de lecteurs.

4 Le diner des cinq Philosophes

Cinq philosophes se trouvent autour d'une table. Chacun des philosophes a devant lui un plat de spaghetti et deux fourchettes. Un philosophe n'a que trois états possibles :

- penser pendant un temps indéterminé,
- être affamé (pendant un temps déterminé et fini sinon il y a famine),
- manger pendant un temps déterminé et fini.

Des contraintes extérieures s'imposent à cette situation :

- quand un philosophe a faim, il va se mettre dans l'état "affamé" et attendre que les fourchettes soient libres ;
- pour manger, un philosophe a besoin de deux fourchettes : celle qui se trouve à gauche de sa propre assiette, et celle qui se trouve à gauche de celle de son voisin de droite (c'est-à-dire les deux fourchettes qui entourent sa propre assiette) ;
- si un philosophe n'arrive pas à s'emparer d'une fourchette, il reste affamé pendant un temps déterminé, en attendant de renouveler sa tentative.

Le problème consiste à trouver un ordonnancement des philosophes tel qu'ils puissent tous manger, chacun à leur tour.