La concurrence

Exemple

- Lecture / Ecriture
- Soit A une donnée et 2 processus ayant le même code
  - Lire(A)
  - A$\leftarrow$ A+1
  - Ecrire(A)
- Donner les exécutions possibles, que vaut A pour chaque exécutrion ?
- Quel est le problème ?

Section Critique

Portion de code qui doit être exécuté en exclusion mutuelle

- Problème d'exclusion mutuelle : 2 propriétés :

Sûreté : Un **seul** processus peut entrer en section critique
Vivacité : Un processus demandant la section critique finira par l'obtenir
Efficacité (ne fait pas partie de la définition stricte)

Solutions

- Désactivation des interruptions
- Variable de verrou
- Alternance stricte
- Algorithme de Peterson

Algorithme de Peterson

while *vrai* **do**
  Actions avant la S .C.
  $D_i \longleftarrow vrai$
  $Tour \longleftarrow i$
  while $(D_{((i+1)\%2)}) \cap (Tour = i)$ **do**
    Rien
  end while
  Actions de la S.C.
  $D_i \longleftarrow faux$
  Actions suivant la S.C.
end while

Les Sémaphores

Un sémaphore S (le plus simple) est un drapeau qui est :

- soit levé
- soit baissé

Avant d'entrer en S.C, un processus attend que le drapeau soit levé, puis le baisse : opération $P()$ A la sortie de la S.C., le processus lève le drapeau opération $V()$ et le S.E réveille un processus en attente

Les Sémaphores

Les opérations P() et V() sont atomiques

- $P()$: le test et la décrémentation sont **atomiques**.
- $V()$: l'incrémentation est atomique

2 opérations atomiques ne peuvent être exécutées simultanément
En aucun cas, 2 processus ne verront le drapeau levé et l'abaisseront

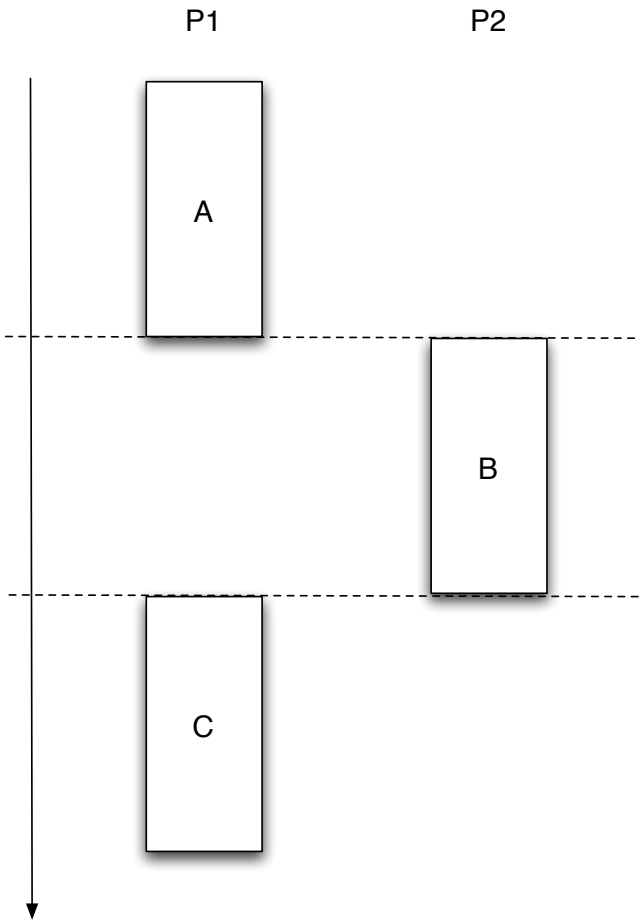
Les Sémaphores (System V)

- S entier entre 0 et n
- $P(n)$: bloquante si $S < n$ sinon décrémente S de n.
- $V(n)$: incrémente S de n

Exemple : Problème du producteur-consommateur

- 2 processus partagent un buffer commun
  - le producteur place des informations
  - le consommateur les retire
- Les problèmes :
  - buffer est plein et le producteur veut placer une information
  - buffer est vide et le consommateur veut retirer une information
- Ecrire une solution à l'aide de sémaphores

Ordonnancement à l'aide de sémaphores



Ordonnancement à l'aide de sémaphores

- Comment réaliser cette exécution à l'aide de sémaphore ?
  - Combien de sémaphores doit-on utiliser ?
  - A quelle(s) valeur(s) doit on les initialiser ?
- Solution
  - 2 sémaphores S_1 et S_2 initialisés à 0.
  - Où placer les opérations $P()$ et $V()$?
- $P1.A, P1.V(S_1), P1.P(S_2), P1.C$ et $P2.P(S_1), P2.B, P2.V(S_2)$
- Détailler l'exécution

Ordonnancement à l'aide de sémaphores

- Comment réaliser cette exécution à l'aide de sémaphore ?
  - Combien de sémaphores doit-on utiliser ?
  - A quelle(s) valeur(s) doit on les initialiser ?
- Solution
  - 2 sémaphores S_1 et S_2 initialisés à 0.
  - Où placer les opérations $P()$ et $V()$?
- $P1.A, P1.V(S_1), P1.P(S_2), P1.C$ et $P2.P(S_1), P2.B, P2.V(S_2)$
- Détailler l'exécution

Ordonnancement à l'aide de sémaphores

- Comment réaliser cette exécution à l'aide de sémaphore ?
  - Combien de sémaphores doit-on utiliser ?
  - A quelle(s) valeur(s) doit on les initialiser ?
- Solution
  - 2 sémaphores S_1 et S_2 initialisés à 0.
  - Où placer les opérations $P()$ et $V()$?
- $P1.A, P1.V(S_1), P1.P(S_2), P1.C$ et $P2.P(S_1), P2.B, P2.V(S_2)$
- Détailler l'exécution

Ordonnancement à l'aide de sémaphores

- Comment réaliser cette exécution à l'aide de sémaphore ?
  - Combien de sémaphores doit-on utiliser ?
  - A quelle(s) valeur(s) doit on les initialiser ?
- Solution
  - 2 sémaphores S_1 et S_2 initialisés à 0.
  - Où placer les opérations $P()$ et $V()$?
- $P1.A, P1.V(S_1), P1.P(S_2), P1.C$ et $P2.P(S_1), P2.B, P2.V(S_2)$
- Détailler l'exécution

Exercice

Soit un système sur lequel on exécute trois processus P_1, P_2 et P_3. P_1 exécute successivement du code découpé en trois parties A, B et C. P_2 exécute du code découpé en deux parties D et E. P3 ne doit commencer réellement à s'exécuter que lorsque P_1 aura terminé la partie A. De plus B et D doivent être en exclusion mutuelle.

- Construire le graphe de précédence de ce système de processus.
- Résoudre le problème entre P_3 et P_1 avec un sémaphore.
- Même chose pour l'exclusion mutuelle entre P_1 et P_2. Peut-on utiliser le même sémaphore ?
- Donner le code de chacun des processus.

Et sinon ?

Que faire lorsque les processus n'ont pas accès à une mémoire partagée commune ?

- Communication entre les processus
  - Tubes
  - Sockets

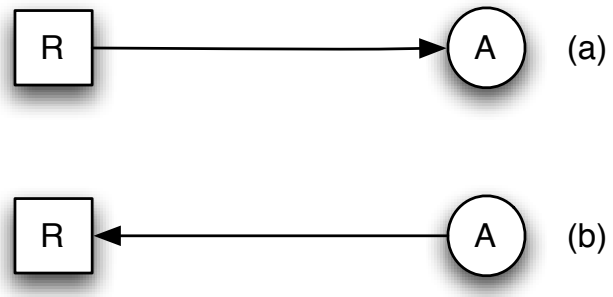
Interblocage

Soit 2 processus et 2 sémaphores S1 et S2 initialisés à 1 :

- P1
  - P(S1)
  - P(S2)
  - Section critique
  - V(S1)
  - V(S2)
- P2
  - P(S2)
  - P(S1)
  - Section critique
  - V(S2)
  - V(S1)

Y-a-t'il moyen de bloquer le système ?

Un outil pour la détection d'interblocage : Graphe des ressources



(a) A détient la ressource R

(b) A demande la ressource R

Un interblocage se produit lorsqu'un circuit se forme dans le graphe des ressources.
Ceci permet une détection des interblocages mais en aucun cas une prévention.

Solution

Tuer un processus qui contribue à l'interblocage.
Choix du processus : Critères ?

- Temps CPU consommé ?
- Ressources attachées ?
- Au centre de combien d'interblocage ?

Beaucoup de travaux, mais peu de résultats convaincants