Systèmes d'exploitation
TD 3: Interblocages.

Frédéric Vivien, bureau 306, e-mail: fvivien

4 mars 1997

1 Détection des interblocages

1.1 Une seule ressource de chaque type

Nous considérons ici un système qui ne contient qu'une seule ressource de chaque type. Imaginer un algorithme qui détecte si un ensemble de processus est ou non en situation d'interblocage.

Application

Le système suivant est-il en situation d'interblocage?

1. Le processus A détient la ressource R et veut la ressource S.
2. Le processus B ne détient aucune ressource mais veut la ressource T.
3. Le processus C ne détient aucune ressource mais veut la ressource S.
4. Le processus D détient la ressource U et veut les ressources S et T.
5. Le processus E détient la ressource T et veut la ressource V.
6. Le processus F détient la ressource W et veut la ressource S.
7. Le processus G détient la ressource V et veut la ressource U.

1.2 Plusieurs ressources de chaque type

Même question quand il peut coexister plusieurs ressources du même type.

Application

Le système suivant est-il en interblocage? Si le processeur B requière 4 ressources de type 2 et 2 ressources de type 3, sa requête peut-elle être satisfaite immédiatement?

Ressources du système:

Type de ressource	Nombre de ressources
1	3
2	14
3	12
4	12

Ressources détenues:

Processus	Ressource 1	Ressource 2	Ressource 3	Ressource 4
A	0	0	1	2
B	1	0	0	0
C	1	3	5	4
D	0	6	3	2
E	0	0	1	4

Ressources désirées:

Processus	Ressource 1	Ressource 2	Ressource 3	Ressource 4
A	0	0	0	0
B	0	7	5	0
C	1	0	0	2
D	0	0	2	0
E	0	6	4	2

2 Évitement des interblocages: algorithme du banquier

2.1 Un seul type de ressources

cf. tableau...

Application

Parmi les trois états suivants d'un système avec dix copies de la ressource, lesquels sont en interblocage, et lesquels ne le sont pas?

	Détient	*Demande*
A	0	6
B	0	5
C	0	4
D	0	7

	Détient	*Demande*
A	1	6
B	1	5
C	2	4
D	4	7

	Détient	*Demande*
A	1	6
B	2	5
C	2	4
D	4	7

2.2 Plusieurs types de ressources

Imaginer comment étendre l'algorithme du banquier au cas ou plusieurs types de ressources coexistent.

Application

Appliquer l'algorithme proposé à l'exemple ci-dessous. Que se passe-t-il si le système accorde une ressource de type 3 au processus B et que le processus E réclame à son tour une telle ressource?

Ressources du système:

Type de ressource	Nombre de ressources
1	6
2	3
3	4
4	2

Ressources détenues:

Processus	Ressource 1	Ressource 2	Ressource 3	Ressource 4
A	3	0	1	1
B	0	1	0	1
C	1	1	1	0
D	1	1	0	0
E	0	0	0	0

Ressources désirées:

Processus	Ressource 1	Ressource 2	Ressource 3	Ressource 4
A	1	1	0	0
B	0	1	1	2
C	3	1	0	0
D	0	0	1	0
E	2	1	1	0

3 Exercices

3.1 Séparation de biens

"Cendrillon et le prince charmant divorcent. Pour partager leurs biens, ils ont adopté l'algorithme suivant. Chaque matin, chacun envoie une lettre à l'avocat de l'autre pour demander un bien. Comme il faut une journée pour distribuer le courrier, ils ont convenu que s'ils découvrent qu'ils ont, tous les deux, demandé le même jour le même bien, ils envoient le jour suivant une lettre qui annule la demande. Leurs biens comportent leur chien Woofer, sa niche, leur canari Tweeter, et la cage de ce dernier. Les animaux sont habitués à leurs "maisons". Toute répartition qui sépare un animal de sa "maison" n'est pas valide, et toute l'opération doit être reprise à partir du début. Cendrillon et le Prince désirent tous les deux à tous prix Woofer. Comme ils doivent prendre des vacances (séparées), chacun des conjoints a programmé un ordinateur personnel pour mener à bien la négociation. Quand ils rentrent de vacances, les ordinateurs négocient toujours. Pourquoi ? Les interblocages sont-ils possibles ? La famine est-elle possible ?" (Andrew Tanenbaum, Les systèmes d'exploitations)

3.2 Utilisation d'un pont

Un pont est trop étroit pour permettre à deux véhicules de se croiser. Écrire, à l'aide de sémaphores, un programme qui régie la circulation sur ce pont tout en évitant interblocages et famines.