

**PARTAGE DE RESSOURCES**

Il est recommandé d'utiliser un tableur pour réaliser les différents calculs. On rappelle que les supports sont disponibles à <http://mathieu.delalandre.free.fr/teachings/operating2.html>

**1. Acquisition de ressources, graphe d'allocation**

On considère un problème de partage de ressources R0, R1 et R2, existant en une seule instance, entre les processus P0, P1 et P2. Le tableau suivant indique, pour chacun des processus P0, P1 et P2, les capacités C, les dates de requête Q(t) sur les ressources R0, R1 et R2 (exprimées à partir des dates « s » de premières exécutions des processus) et les délais d'utilisation U de ces ressources (i.e. pour une ressource donnée,  $Q(t)+U \leq s+C$ ). Ces requêtes sont réalisées par synchronisation par sémaphores de type mutex, les opérations « down » et « up » sont donc respectivement exécutées aux instants Q(t) et Q(t)+U.

	C	R0		R1		R2	
		Q(t)	U	Q(t)	U	Q(t)	U
<b>P0</b>	16	s+6	9	Na	Na	s+9	2
<b>P1</b>	18	Na	Na	s+9	7	s+5	8
<b>P2</b>	8	s+5	2	s+1	5	Na	Na

Considérant le schéma d'ordonnancement ci-dessous, déterminez sur l'intervalle de temps considéré  $t = [0-41]$  les informations suivantes : graphes d'allocation de ressources (a) à (h), états des piles des sémaphores / mutex S0, S1 et S2 (gérant respectivement l'accès aux ressources R0, R1 et R2), états des processus P0, P1 et P2, allocation des ressources.

<b>Burst</b>	9	7	1	2	5	7	4	7
<b>Processus</b>	P1	P0	P2	P0	P1	P0	P1	P2
<b>Événement</b>	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h,i)

Répétez la démarche considérant le schéma d'ordonnancement ci-dessous sur l'intervalle de temps  $t = [0-23]$ . Expliquez alors les phénomènes observés aux événements (e) et (f).

<b>Burst</b>	7	7	3	2	2	2
<b>Processus</b>	P1	P0	P2	P0	P2	P1
<b>Événement</b>	(a)	(b)	(c)	(d)	(e)	(f)

**2. Approche basée sur les états saufs/non saufs,**

2.1. Etats saufs/non saufs

Soit un système où deux ressources R0, R1 sont partagées par 4 processus P0, P1, P2 et P3. Au sein de ce système, R0 et R1 existent en 20 et 15 instances respectivement. Les besoins en ressources R0 sont de 8, 2, 8 et 9 et ceux en ressources R1 sont de 2, 9, 7 et 5 pour P0, P1, P2 et P3 respectivement. A un instant  $t_0$ , on considère la matrice d'allocation suivante entre P0, P1, P2, P3 et R0, R1. Déterminez si ce système est en état sauf ou non sauf. Si sauf, indiquez la séquence de désallocation envisagée permettant de satisfaire l'ensemble des processus.

	R0	R1
P0	5	2
P1	2	3
P2	4	5
P3	6	1

## 2.2. Diagramme de progression jointe

On considère un cas où deux processus P0, P1 se partagent 3 ressources R0, R1 et R2. Le tableau suivant indique, pour chacun de ces deux processus, les capacités, les dates de requête Q(t) sur les ressources (exprimées à partir des dates « s » de premières exécutions des processus) et les délais d'utilisation U.

	C	R0		R1		R2	
		Q(t)	U	Q(t)	U	Q(t)	U
P0	20	s+7	6	s+6	6	s+3	14
P1	20	s+8	8	s+7	4	s+12	5

A partir de ces informations, tracez le diagramme de progression jointe. Vous devrez calculer les dates de libération des ressources  $R(t) = Q(t) + U$  pour chacun des processus et ordonner les Q(t), R(t) par processus. Déterminez les zones d'exclusions mutuelles, les zones d'états saufs et non saufs et la situation d'interblocage. Finalement, tracez un ou plusieurs chemins dans le diagramme de progression jointe permettant d'arriver à l'interblocage identifié, vous pourrez donner les graphes d'allocation de ressources correspondant.

## 2.3. Rejet d'allocation de processus

On considère un système où la gestion des ressources se réalise via un protocole de rejet d'allocation de processus. Au sein de ce système, 3 ressources R0, R1 et R2 sont disponibles en 7 instances maximum chacune. A un instant  $t_0$  donné, on considère que 3 processus P0, P1 et P2 ont été alloués au sein du système. Les matrices A, C suivantes décrivent les états d'allocation « (A)llocated » et de réclamation « (C)laim » de ces processus.

A	R0	R1	R2
P0	1	0	2
P1	1	0	1
P2	2	3	0

C	R0	R1	R2
P0	1	0	3
P1	1	0	2
P2	3	3	0

Vérifiez dans un premier temps l'état d'allocation du système à l'instant  $t_0$ . Vous pourrez pour cela calculer le vecteur V et vérifier si les contraintes sont respectées entre V, R, A et C.

Soit le cas où trois processus PA, PB et PC demandent l'accès au système à  $t > t_0$ . La matrice ci-dessous donne les réclamations en ressource de ces trois processus. On considérera pour  $\forall t > t_0$  le maintien de l'allocation de P0, P1, P2 dans le système. Considérant  $Q_A$ ,  $Q_B$  et  $Q_C$  les requêtes correspondants aux processus PA, PB et PC, indiquez l'état d'allocation du système pour les ordres d'arrivé suivants (1)  $Q_A, Q_B, Q_C$  (2)  $Q_B, Q_C, Q_A$  (3)  $Q_A, Q_C, Q_B$ .

C	R0	R1	R2
PA	2	1	1
PB	1	3	2
PC	1	1	0

#### 2.4. Algorithme de détection d'interblocages

Soit un système où la gestion des ressources est assurée par détection d'interblocages et récupération. Le protocole de détection mis en place se base sur la détection des états saufs / non saufs. A un instant  $t_0$ , on observe la situation où 3 processus P0, P1 et P2 se partagent 3 ressources R0, R1 et R2. Ces ressources R0, R1 et R2 existent en 9, 9 et 12 instances maximum au sein du système, respectivement. Les matrices A, N suivantes décrivent les états d'allocation « (A)llocated » et besoin « (N)eeded » des processus P0, P1 et P2 à  $t_0$ .

A	R0	R1	R2
P0	2	1	2
P1	1	1	2
P2	3	4	5

N	R0	R1	R2
P0	4	3	5
P1	1	3	2
P2	6	4	7

Vérifiez dans un premier temps l'état d'allocation du système à l'instant  $t_0$ . Vous pourrez pour cela calculer le vecteur V et vérifier que les contraintes soient respectées entre V, R, A et N.

Dans un second temps, déroulez l'algorithme de détection en précisant pour chacune des étapes les valeurs des vecteurs W, F et l'ordre dans lequel les ressources sont allouées aux processus. Précisez alors si le système est dans un état sauf ou non.

#### 2.5. Algorithme du banquier

Soit un système au sein duquel la gestion de ressources est assurée à l'aide de l'algorithme du banquier. Au sein de ce système, deux ressources R0, R1 existent en 8 et 4 instances maximum respectivement. A un instant  $t_0$ , on observe la situation où 3 processus P0, P1 et P2 se partagent les ressources. Les matrices A, C suivantes décrivent les états d'allocation « (A)llocated » et de réclamation « (C)laim ».

A	R0	R1
P0	2	3
P1	3	0
P2	1	1

C	R0	R1
P0	3	4
P1	3	1
P2	2	1

Vérifiez dans un premier temps l'état d'allocation du système à l'instant  $t_0$ . Vous pourrez pour cela calculer le vecteur V et vérifier si les contraintes sont respectées entre V, R, A et C.

Déterminer ensuite si la requête  $Q_0 = (1,0)$  pour P0 conduit à un état sauf ou non sauf du système en vous appuyant sur les algorithmes « SA » et « RRA ».