

Examen semestriel

Module : Systèmes d'exploitation

Corrigé

Exercice 1 : Expliquez brièvement la méthode d'allocation indexée. Quels sont ses avantages ? ses inconvénients ?

Réponse :

Dans la méthode d'allocation indexée, il faut réserver un bloc index qui contiendrait tous les numéros de blocs du fichier. Ainsi, pour accéder au fichier, on accède d'abord à l'index; on récupère les n° de blocs du fichier; et on accède par la suite aux blocs.

(2 points)

Avantages :

Rapidité d'accès : L'utilisation de l'index nous permet de connaître rapidement les n° de blocs composant le fichier.

(2 points)

Inconvénients :

Perte d'espace : L'espace occupé par le bloc index lui-même.

(2 points)

Exercice 2 : On considère 3 types de processus A, B et C. Ces processus utilisent une ressource unique R. La ressource R peut être utilisée simultanément par un nombre quelconque de processus, mais au maximum de deux types différents. Par exemple, 5 processus A et 7 processus B peuvent utiliser en même temps la ressource R, mais si des processus C veulent utiliser la ressource R, ils doivent attendre la fin de tous les processus A ou bien la fin de tous les processus B.

Proposez une solution à ce problème de synchronisation en utilisant les sémaphores (soignez vos déclarations et votre code).

Réponse :

Déclaration des variables :

nA : Nbre de processus A voulant accéder à la ressource A (init à 0).

nB : Nbre de processus B voulant accéder à la ressource A (init à 0).

nC : Nbre de processus C voulant accéder à la ressource A (init à 0).

semA : sémaphore d'exclusion mutuelle (init à 1) pour protéger la variable partagée nA

semB : sémaphore d'exclusion mutuelle (init à 1) pour protéger la variable partagée nB

semB : sémaphore d'exclusion mutuelle (init à 1) pour protéger la variable partagée nC

W : sémaphore (init à 2) pour bloquer l'accès des processus.

(2 points)

<pre> Processus A Wait (semA) nA:= nA + 1 si nA = 1 Alors wait(W) finsi signal(semA) // Accéder à la Ressource R Wait (semA) nA:= nA - 1 si nA = 0 Alors signal(W) finsi signal(semA) </pre>	<pre> Processus B Wait (semB) nB:= nB + 1 si nB = 1 Alors wait(W) finsi signal(semB) // Accéder à la Ressource R Wait (semB) nB:= nB - 1 si nB = 0 Alors signal(W) finsi signal(semB) </pre>	<pre> Processus C Wait (semC) nC:= nC + 1 si nC = 1 Alors wait(W) finsi signal(semC) // Accéder à la Ressource R Wait (semC) nC:= nC - 1 si nC = 0 Alors signal(W) finsi signal(semC) </pre>
--	--	--

(12 points)