

Examen de Système d'exploitation

Nature de l'épreuve : DC D.S. E.F.
Date de l'épreuve : 06/01/2024
Durée de l'épreuve : 01H30
Sections : LIRS 1 - LGSI 1

Documents : autorisés non autorisés
Calculatrice : autorisée non autorisée
Session : principale contrôle
Barème approximatif : 6-10-4

- NB :**
- Rédigez soigneusement les solutions en utilisant les mêmes conventions vues en cours. La clarté et la concision des réponses sont des éléments importants d'appreciation.
 - Il est strictement interdit d'écrire au crayon noir ou d'utiliser le correcteur.
 - Tous les exercices sont indépendants.
 - Lire attentivement et entièrement chaque exercice avant de commencer la résolution.
 - Numéroter les feuilles de réponse à l'examen.

[.../6 points]

Exercice 1 : Questions à courte réponse

- 1) On considère trois processus qui nécessitent respectivement 10, 20 et 30 unités de temps CPU et arrivent respectivement aux instants 0, 2 et 6. Quel est le nombre de commutations de contexte nécessaires si le système d'exploitation utilise l'algorithme d'ordonnancement SRTF ? Justifier votre réponse.

- 2) On considère les propositions suivantes concernant les threads implémentés au niveau utilisateur et les threads implémentés au niveau du noyau. Pour chacune des affirmations suivantes, indiquez, en justifiant votre réponse, si elles sont correctes ou fausses.
- a) Le temps de changement de contexte est plus long pour les threads implémentés au niveau du noyau que pour les threads implémentés au niveau utilisateur.

Correcte

- b) Les threads implémentés au niveau utilisateur n'ont besoin d'aucun support matériel.
Fausse / Les threads implémentés au niveau utilisateur besoin d'un support matériel
- c) Les threads implémentés au niveau du noyau peuvent être affectés à différents processeurs d'un système multiprocesseur.
- Fausse / les threads implémentés au niveau du noyau ne peuvent pas être affectés à différents processeurs d'un système multiprocesseur car chaque processeur a ses propres threads
- d) Le blocage d'un thread implémenté au niveau du noyau bloque tous les threads associés.

Correcte

J) Soient les deux processus (codes des fonctions écrits en langage C) suivants notés P1 et P2, qui s'exécutent en concurrence et qui partagent une variable B initialement à 2 :

P1()

{ C = B - 1;
B = 2 * C;

P2()

{ D = 2 * B;
B = D - 1;

}

Quelles sont les valeurs possibles (envisager tous les cas possibles) que peut prendre la variable partagée B après la fin de l'exécution de P1 et P2 ?

pour P1 :

C = 2 - 1 = 1
B = 2 * 1 = 2

B = 2

pour P2 :

D = 2 * 2 = 4
B = 4 - 1 = 3

B = 3

Exercice 2 : L'ordonnancement EDF (Earliest Deadline First) [...] /10 points

L'ordonnancement EDF (Earliest Deadline First) est un algorithme d'ordonnancement temps réel de processus. A chaque fois qu'un processus demande du temps CPU, il doit préciser une échéance (une date limite > 0). Le processus doit obtenir son temps CPU avant d'atteindre la date limite. L'ordonnanceur vise ainsi à satisfaire les demandes avant leurs échéances. Pour se faire, il gère une liste des processus prêts, classés par ordre croissant des échéances. L'algorithme exécute le premier processus de la liste qui correspond à celui dont l'échéance est la plus proche.

Si deux processus ont la même échéance, l'ordonnanceur sélectionne le processus le plus prioritaire. Lorsqu'un processus devient prêt ou arrive dans le système, l'ordonnanceur vérifie si son échéance est antérieure ou égale à celle du processus en cours d'exécution. Le processus en cours est préempté pour l'une des raisons suivantes :

- L'échéance du processus en cours est plus loin que celle d'un processus prêt.
- L'échéance du processus en cours est la même que celle d'un processus prêt plus prioritaire.

On considère les trois processus A, B et C suivants :

- A demande le CPU toutes les 20 ms et requiert à chaque fois 4 ms de temps CPU.
- B demande le CPU toutes les 30 ms et requiert à chaque fois 10 ms de temps CPU..
- C demande le CPU toutes les 40 ms et requiert à chaque fois 20 ms de temps CPU.

A est supposé plus prioritaire que B qui est, à son tour, plus prioritaire que C.

On suppose qu'au départ, les processus sont prêts et que l'échéance de chaque demande est la date de la prochaine demande. Par exemple, pour A, les dates d'échéances sont successivement 20, 2*20, 3*20,

- 1) Donnez le diagramme de Gantt pour les 90 premières ms. Remplir pour répondre le tableau suivant :

TPS	Process En cours	Durée		Echéances			File d'attente	Justifications et commentaires
		A	B	C				
0	A	4	20	30	40	B - C	Fin de A	
5	B	10	16	20	40	A - C	Fin de B	
15	C	20	16	20	20	A - B	Fin de C	
35	A	4	12	20	20	B - C	Fin de A	
39	B	10	12	10	20	A - C	Fin de B	
48	C	20	12	10	0	A - B	Fin de C	
68	A	4	8	10	0	B	Fin de A	
73	B	10	8	0	0	A	Fin de B	
83	A	4	4	0	0	A	Fin de A	
87	A	4	0	0	0	-	Fin de A	
91	-	-	-	-	-	-	-	-
101	-	-	-	-	-	-	-	-
111	-	-	-	-	-	-	-	-

Exercice 3 : Vérification de la cohérence d'un disque

[.../4 points]

- On souhaite écrire un algorithme permettant la vérification de la cohérence d'un disque.
1. Définir les structures des données nécessaires à cet algorithme.
 2. Ecrire un algorithme détaillé permettant de définir les actions à entreprendre pour tous les cas d'incohérences possibles.