ORDRE DES INGÉNIEURS DU QUÉBEC

SESSION DE MAI 2018

Toute documentation permise Calculatrices : modèles autorisés seulement Durée de l'examen : 3 heures

14-IF-A5 Systèmes d'exploitation

QUESTION 1 [20 points]

Concepts et services des systèmes d'exploitation

(a) [8 points]

i. La transition du mode usager au mode noyau est une section délicate d'un système d'exploitation. Expliquer ce qui se passe lorsqu'un événement déclenche cette transition.
ii. Dans la majorité des ordinateurs modernes, il y a toujours au moins une partie de code écrite en assembleur qui gère les évènements (interruptions). Pourquoi?

(b) [4 points]

Expliquer le concept de la réentrante.

(c) [8 points]

Parmi les instructions suivantes, quelles sont celles qui sont protégées (qui peuvent être exécutées seulement en mode noyau) ?

i. Modifier le compteur de programme (« *program counter* »)
ii. Modifier le pointeur de pile (« *stack pointer* »)
iii. Modifier le registre qui contrôle le mode noyau/utilisateur
iv. Accès direct aux périphériques d'entrées/sorties

QUESTION 2 [20 points]

Processus, fils et synchronisation

(a) [4 points]

Considérer le bout de code suivant :

```
for (i=0 ; fork() ; i++) {
    if (i==6) {break ;}
    printf(''PID : %d, i = %d\n'', getpid(), i) ;
}
```

Supposez que le numéro d'identification initial du processus est 1234 et que le numéro de processus augmente de 1 à chaque nouveau processus. Supposez aussi qu'il n'y a pas de nouveau processus créé durant l'exécution de ce programme.

(i) Quelles sont les sorties possibles du programme ?
(ii) Pour toute la durée de vie du programme, combien d'appels à fork() sont faits ?

(b) [6 points]

La majorité des systèmes d'exploitation modernes peuvent gérer les fils d'exécution (« *threads* ») et les processus.

i. Qu'est-ce qu'un processus ?
ii. Qu'est-ce qu'un fil d'exécution (« *thread* ») ?

(c) [4 points]

Est-ce qu'un programme qui contient une condition concurrente va toujours se retrouver avec des données corrompues ou n'importe quel autre mauvais fonctionnement ? Expliquer.

(d) [6 points]

Expliquer les termes suivants :

i. Le bloc de contrôle du processus (« *process control block* »)
ii. Les fils de processus en mode usager (« *user-level threads* »)
iii. Les ensembles de fils d'exécution (« threads pool »)

QUESTION 3 [20 points]

Ordonnancement des processus

(a) [3 points]

Dans quelles circonstances l'ordonnanceur de type FIFO va obtenir le plus petit temps de réponse moyen ? Expliquer brièvement.

(b) [3 points]

Dans quelles circonstances l'approche tourniquet (« round-robin ») se comporte de la même façon qu'un FIFO ?

(c) [2 points]

Expliquer comment le vieillissement (« aging ») permet de résoudre le problème de famine.

(d) [12 points]

Considérer les processus suivants :

Processus	Temps de départ (ms)	Durée (ms)
A	0	20
B	4	10
C	10	25
D	17	10
E	25	5

Pour chacune des stratégies d'ordonnancement suivantes, vous devez :

1. Dessiner le diagramme temporel (« Gantt chart »)

2. Calculer le temps d'attente moyen

   i. La tâche la plus courte en premier (*Shortest Job first, SJF*), sans préemption.
   ii. La tâche la plus courte en premier (*Shortest Job first, SJF*), avec préemption.
   iii. Type tourniquet (Round-Robin) avec une tranche de vie (quantum) de 8 ms. Supposer que les nouveaux processus sont toujours ajoutés à la fin de la liste des processus en attente.

QUESTION 4 [20 points]

Interblocage

(a) [8 points]

Considérer les trois bouts de codes suivants:

Thread 1:
Lock(L1)
Lock(L2)
// critical section
Unlock(L2)
Unlock(L1)

Thread 2:
Lock(L3)
Lock(L1)
// critical section
Unlock(L1)
Unlock(L3)

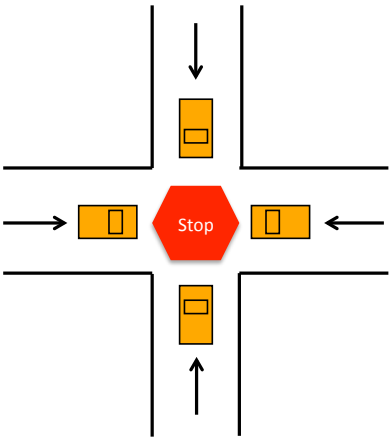
Thread 3:
Lock(L2)
Lock(L3)
// critical section
Unlock(L3)
Unlock(L2)

Au départ, tous les verrous sont initialisés à “non verrouillés”. Supposer que les threads peuvent être exécutés avec n’importe quel entrelacement possible.

Est-ce que ce code est problématique? Si oui, montrer le et expliquer comment le problème peut être résolu si on considère que l’utilisation de trois verrous et obligatoire. S’il n’y a pas de problème, expliquer pourquoi.

(b) [12 points]

Considérer la situation suivante où des signes « arrêt » sont présents dans toutes les directions



i. Est-ce que la règle « laisser passer la voiture de droite » empêche la création d’un interblocage? Expliquez.
ii. Est-ce que les 4 conditions nécessaires pour produire un interblocage sont présentes dans cet exemple ? Expliquer.
iii. Quelle est la différence entre interblocage et famine ?

QUESTION 5 [20 points]

Gestion de la mémoire

(a) [12 points]

Pour chacune des paires de termes suivants, définir chacun des termes et donner les différences clés entre les termes.

i. Page et cadre (« *frame* »)
ii. Pagination et segmentation
iii. Bit de référence ou bit de modification (« *dirty bit* »)
iv. Fragmentation interne et externe

(b) [8 points]

Un système de gestion de mémoire paginée contient 4 cadres. Si nous avons la séquence de pages suivantes :

{ a, b, c, d, e, d, c, b, a, b, c, d, e, f, d, c, b, a, b, c, d, e, f, g, f, e, d, c, b, a }

i. En utilisant l'algorithme FIFO pour le remplacement de page, donner le contenu de la mémoire physique après chaque référence mémoire. Donner aussi le nombre de fautes de page.

ii. En utilisant l'algorithme LRU (*Least Recently Used*) pour le remplacement de page, donner le contenu de la mémoire physique après chaque référence mémoire. Donner aussi le nombre de fautes de page.