ORDRE DES INGÉNIEURS DU QUÉBEC

SESSION DE MAI 2019

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) [4 points]

Qu'est-ce qu'un système d'exploitation temps réel ?

(b) [5 points]

Qu'est-ce qu'un noyau (*kernel*) ?

(c) [5 points]

Quel est le principal objectif de la multiprogrammation ?

(d) [6 points]

Nommer et décrire trois différents types de mémoires.

QUESTION 2 [20 points]

Processus, fils et synchronisation

(a) [6 points]

Qu'est-ce qu'un fil d'exécution (« thread ») ? Expliquez en détails.

(b) [10 points]

Considérer le programme suivant qui contient 2 threads pouvant être exécuté en parallèle:

Initialisation

```
int a = 5 ;
Int b = 0 ;
Int c = 0 ;
```

Thread 1

```
 if(a<0){
      c = b – a
  } else{
      c = b + a
  }
```

Thread 2

```
    b  = 10 ;
    c  = -3 ;
```

Quelles sont les valeurs possibles de c lorsque le programme se termine ? Supposer que la lecture et l'écriture des variables sont des opérations atomiques.

(c) [4 points]

Combien de fois le programme suivant imprime le mot « Bonjour ! » ?

```
void main(int argc, char** argv)
{
    int i;
    for(i=0 ; i<2 ; i++)
        fork() ;
    printf( ''Bonjour !\n'') ;
}
```

QUESTION 3 [20 points]

Ordonnancement des processus

(a) [4 points]

Est-ce qu'implémenter le concept d'affinité du processeur dans un ordonnanceur pour multiprocesseur permettra vraisemblablement de réduire le nombre de « miss » dans les caches. Expliquer.

(b) [4 points]

Définir le concept de balancement de la charge (« *load balancing* ») et dans quels contextes c'est important.

(c) [12 points]

Soit les processus suivants :

Processus	Temps d'arrivée (ms)	Durée (ms)
A	0	30
B	10	5
C	15	15
D	20	5
E	25	10

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

Dessiner le diagramme temporel (« Gantt chart »)

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 5 ms. Les nouveaux processus sont placés au début de la file (passe devant les processus existants dans l'ordre d'exécution).

QUESTION 4 [20 points]

Interblocage

(a) [2 points]

Qu'est-ce qu'un un « état sûr » ?

(b) [10 points]

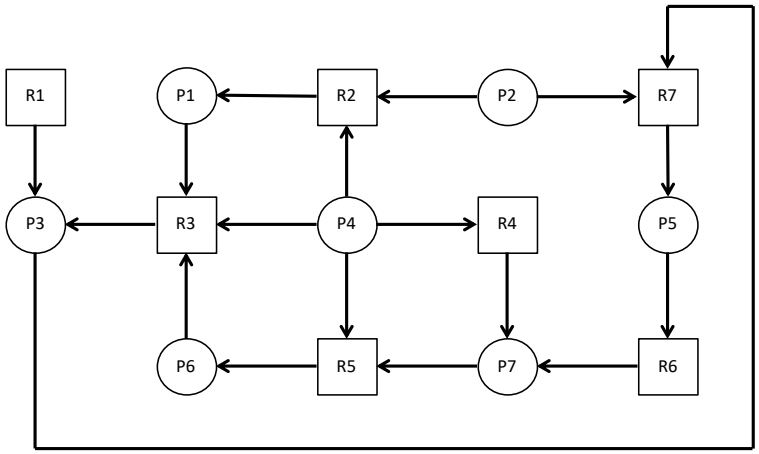
Considérez la situation suivante:

	Alloués				Besoins				Disponibles			
	A	B	C	D	A	B	C	D	A	B	C	D
P1	1	0	0	1	2	1	2	3	2	0	2	1
P2	1	0	1	1	2	0	1	0				
P3	1	1	0	0	2	2	1	1				
P4	1	2	0	1	2	0	1	2				

i) Utilisez l'algorithme du banquier pour déterminer si le système est dans un état sûr ainsi que la séquence d'exécution sûre, s'il en existe une.
ii) Si le système est dans un état sûr, donnez l'ordre d'exécution des processus qui évitera un interblocage.

(c) [8 points]

Voici le graphe d'allocation de ressources présentant l'état du système à un instant donné. On suppose que toutes les ressources n'offrent qu'une seule instance.



i) Donner les ressources détenues par chacun des processus du système.

ii) Est-ce qu'il y a un interblocage dans ce système? Si oui, quels sont les processus qui y participent?

iii) Dessiner le graphe d'attente.

QUESTION 5 [20 points]

Gestion de la mémoire

(a) 2 points]

Parmi les choix suivants, quelle adresse est générée par le CPU ?

i) Adresse physique
ii) Adresse absolue
iii) Adresse logique
iv) Aucune de ces réponses

(b) [2 points]

Un programme connaît quel type d'adresse :

i) Adresse physique
ii) Adresse absolue
iii) Adresse logique
iv) Aucune de ces réponses

(c) [2 points]

La table de page contient :

i) Adresse de base de chaque page dans le mémoire physique
ii) Le « page offset »
iii) La taille de la page
iv) Aucune de ces réponses

(d) [2 points]

Un système d'opération maintient une page de table pour :

i) Chaque processus
ii) Chaque « *thread* »
iii) Chaque instruction
iv) Chaque adresse

(e) [12 points]

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

A B C F D A E A C A B D A E F G B D

Pour chacun des algorithmes suivants, donnez le contenu des cadres pour chaque référence, indiquez les références qui causent une faute de page et donnez le nombre de fautes de page total.

i) L'algorithme FIFO
ii) L'algorithme LRU
iii) L'algorithme optimal