



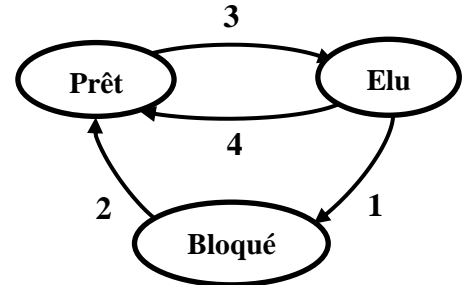
****Examen de Systèmes d'Exploitation 1****

Date : 21/05/2024

Durée: 1h30 - Documentation non autorisée

Exercice 1 : (Questions de Compréhension : 5 pts) (10 minutes)

Q1) Etant donné le diagramme d'états/transitions suivant, citer la/les transition(s) qui doivent être supprimée(s) si on utilise un algorithme d'ordonnancement sans réquisition (non préemptif). Justifier votre réponse.



.....

Q2) Question : Qu'est-ce qu'un déroutement ?

.....

Q3) Quels sont les avantages et inconvénients du choix d'un quantum petit pour l'algorithme de scheduling Round Robin ?

.....

Q4) Décrivez ce qui se passe, du côté du système d'exploitation, lorsqu'une touche de clavier est pressée :

.....

Q5) Dans quel cas est-il intéressant de masquer une interruption ?

.....

Exercice 2 : (Ordonnancement : 5 pts) (25 minutes)

La figure suivante représente le diagramme de Gantt d'un scheduling du processeur utilisant l'algorithme « **Round Robin** » et trois processus : **A, B** et **C**. (Si les processus arrivés en même temps ont file d'attente, le système prend l'ordre d'arrivée des processus **A, B, C**)

A	B	C	A	B	A	inactif	C	
0	3	6	7	10	12	15	16	20

Q1) Quelle est la durée du quantum ?

.....

Q2) Quel est le temps d'attente du processus A ?

.....

Q3) Quel est le temps de réponse du processus B ? Justifiez.

.....

.....

Q4) Quel est le temps d'attente du processus C ? Justifiez.

.....

.....

Q5) Que s'est-il passé entre les instants $t = 15$ et $t = 16$? Justifiez.

.....

.....

.....

Q6) Quel est l'état du processus C à l'instant $t = 9$.

.....

.....

Q7) Dessinez le diagramme de Gantt du même problème, mais en considérant un quantum égal à 4.

.....

.....

.....

Exercice 3 : (La primitive fork : 3 pts) (20 minutes)

Dans cet exercice on suppose que les numéros des **PIDs** attribués aux processus sont strictement croissants. Si un père a un **PID (n)**, son fils aura un **PID (m)** tel que ($m > n$). Le premier processus fils sera créé aura un PID inférieur au seconde processus fils créé ($PID(Père) = n \Rightarrow PID(Fils1) = n+1, PID(Fils2) = n+2, \dots$ etc.).

Soit le code suivant :

```
int main ()
{
    pid_t p1 , p2 ;
    p1 = fork () ;
    p2 = fork () ;
    if ( ( p1- p2 ) > 0 )
        fork () ;
    printf ("Je suis %d : p1 = %d , p2 = %d \n",getpid(), p1, p2) ;
    return 0 ;
}
```

Q1) Donnez le nombre ainsi que l'arborescence des processus créés par ce code.

.....

.....

.....

.....

.....

.....

.....

.....

Q2) Proposez pour chaque processus créé son affichage à l'écran. Nous rappelons que les **PIDs** sont affichés comme suit : *PID-P1=301, PID-P2=302, PID-P3=303, PID-P4=304, ...etc.*

Par exemple, le processus P8 (PID-P8=308) avec p1 (PID-P5) = 305 et p2=0 aura pour affichage :

Je suis 308 : p1 = 305, p2 = 0.

.....
.....
.....
.....
.....
.....

Exercice 4 : (Algorithmes de Remplacement de Pages : 7 pts) (30 minutes)

On considère une mémoire paginée possédant **N** cadres de pages. Soit un programme possédant un espace virtuel de **512 Octets** et la taille de page est de **64 Octets**. Le programme fait référence, durant son exécution, aux adresses virtuelles suivantes :

120, 265, 50, 340, 280, 365, 450, 390, 110, 480, 280, 440, 500, 95

Q1) Donnez le nombre de pages de ce programme ?

.....
.....

Q2) Donnez la suite des numéros de pages référencés ?

.....
.....

Q3) Combien de défauts de pages peuvent se produire au minimum ? Justifiez.

.....
.....
.....

Q4) Combien de défauts de pages peuvent se produire au maximum ? Justifiez.

.....
.....
.....

Sachant que la taille de la mémoire physique est de **320 Octets**.

Q5) Donnez le nombre de cadres (**N**) ?

.....
.....

Q6) En prenant **N=3**, déterminez le nombre de défauts de page générés en applique les algorithmes de remplacement **FIFO**, **LRU**, et **FIFO de la seconde chance**.

FIFO : le nombre de défauts de page =													
Références													
Cadre 1													
Cadre 2													
Cadre 3													
Défaut de page													

LRU : le nombre de défauts de page =													
Références													
Cadre 1													
Cadre 2													
Cadre 3													
Défaut de page													

FIFO de la seconde chance : le nombre de défauts de page =													
Références													
Cadre 1													
Cadre 2													
Cadre 3													
Défaut de page													

Q7) On suppose que le temps de chargement d'une page du disque vers la mémoire est X et que le temps de sauvegarde d'une page de la mémoire vers le disque est Y . Calculer le temps total de traitement (en fonction de X et Y) de la chaîne de références pour l' algorithme **LRU**.

.....

Exercice 5 : (Bonus : 2 pts) (5 minutes)

Si un système utiliser l' algorithme du travail le plus court d'abord pour l'ordonnancement à court terme et une moyenne exponentielle de $\alpha = 0.5$, quel est le prochaine cycle processeur estimé pour un processus aux cycles processeurs **5, 8, 3** et **5** et une valeur initiale de **10** pour e_1 .

.....

Bon courage