

QCM (12,75 pts):Répondre en encerclant **la réponse juste à une question donnée**. Une réponse juste = 0,75 pts, fausse= 0 pt.**Un système d'exploitation est un ensemble de programmes ayant pour objectif :**

1. De gérer les communications entre la machine et l'utilisateur et entre processus
2. **De gérer les ressources physiques et logiques de la machine**
3. De faire exécuter les applications des utilisateurs
4. La réponse juste ne figure pas parmi les réponses précédentes

Un système d'exploitation qui charge en mémoire plusieurs programmes pour les exécuter est appelé:

1. Système mono programmé
2. **Système multiprogrammé**
3. Système multiutilisateur
4. La réponse juste ne figure pas parmi les réponses précédentes

Un processus est défini dans un système d'exploitation comme :

1. Un programme multi-procédure en cours d'exécution
2. Une séquence d'instructions
3. **Une procédure en cours d'exécution**
4. La réponse juste ne figure pas parmi les réponses précédentes

Un processus Zombie est un processus

1. Qui a perdu son père
2. Qui a terminé son exécution en erreur
3. **Qui a terminé son exécution et qui attend la prise en compte de cette fin par son père**
4. Qui a terminé son exécution et son père a pris compte de sa fin

Un thread dans un processus s'exécute dans :

1. **Son propre contexte commun à celui du père**
2. Le contexte de son processus père
3. Son propre contexte différent de celui de son père
4. La réponse juste ne figure pas parmi les réponses précédentes

Un Process Control Bloc (PCB) est une structure de données qui:

1. **Contient toutes les informations concernant un processus**
2. Décrit l'évolution d'un processus dans la machine
3. Contient les informations d'un processus et ses fils
4. La réponse juste ne figure pas parmi les réponses précédentes

Un processus qui fait un calcul puis imprime le résultat passe :

1. De l'état actif à l'état prêt
2. De l'état actif à l'état terminé
3. **De l'état actif à l'état bloqué**
4. La réponse juste ne figure pas parmi les réponses précédentes

Un processus qui vient de lire des données sur écran et ensuite il fera un calcul avec ces données passe :

1. De l'état bloqué à l'état actif
2. De l'état prêt à l'état actif
3. **De l'état bloqué à l'état prêt**
4. La réponse juste ne figure pas parmi les réponses précédentes

L'exécution d'un programme donne toujours lieu à :

1. Un processus
2. **Au moins un processus ou un processus avec au moins un thread**
3. Plusieurs threads
4. La réponse juste ne figure pas parmi les réponses précédentes

Un programme sur disque dur ne peut être chargé en mémoire pour être exécuté que si :

1. Le CPU est libre et il y a de l'espace mémoire pour accueillir le programme
2. **L'espace mémoire est disponible pour accueillir le programme**
3. Le CPU est libre
4. La réponse juste ne figure pas parmi les réponses précédentes

Le rôle d'un ordonnanceur scheduler est d'ordonnancer :

1. L'utilisation de la mémoire.

2. Les opérations d'E/S.
3. Les processus bloqués par les opérations d'E/S.

4. Les processus à exécuter selon un ou plusieurs critères

L'algorithme d'ordonnancement SJF avec réquisition s'appuie en particulier sur Le temps CPU:

1. Consommé par chaque processus
2. **Consommé par le processus actif et le temps CPU du processus nouvellement créé**
3. De tous les processus
4. Consommé par le processus actif ou le temps CPU du processus nouvellement créé

La taille des partitions dans la gestion de la mémoire par partitions fixes est déterminée lors:

1. **De la génération du système**
2. De l'initialisation du système
3. Du chargement des programmes en mémoire
4. La réponse juste ne figure pas parmi les réponses précédentes

La méthode de gestion de la mémoire par partitions fixes est destinée aux systèmes :

1. **De processus industriels**
2. A usage général
3. De type multi-usagers
4. La réponse juste ne figure pas parmi les réponses précédentes

La méthode de gestion de la mémoire par partitions variables a pour objectif principal :

1. D'optimiser le temps d'exécution des programmes
2. **De minimiser la fragmentation interne**
3. D'optimiser le débit de programmes
4. La réponse juste ne figure pas parmi les réponses précédentes

La fragmentation interne est définie comme étant :

1. L'espace des partitions libres de tailles insuffisantes pour pouvoir être allouées
2. **L'espace mémoire resté libre après chargement d'un programme dans une partition**
3. L'espace mémoire demandé par un programme lors de son exécution
4. La partition ne pouvant contenir aucun programme à l'exécution

A l'exécution, le fragment de code suivant génère:

Debut

```
.....
pid_t x ;
x = fork(),
si x= 0 alors
  fork(),
finsi
.....
```

1. Un processus père et un processus fils
2. Un processus père et deux processus fils
3. Un processus fils et un processus petit fils
4. **Un processus père, son fils et un processus petit fils**

Exercice (7,25 points):

Soient un système monoprocesseur multiprogrammé et 3 processus s'exécutant sur ce système. Ces processus P1, P2 et P3 ont la taille 12Ko, 10Ko et 23Ko respectivement. La mémoire de la machine est gérée par la technique partition fixe composée de 10 zones, chacune de capacité 50Ko, 30Ko, 15Ko, 8Ko, 25Ko, 13Ko, 35Ko, 18Ko, 45Ko, 40Ko. On suppose que l'exécution de chaque processus se compose de temps CPU et d'une demande d'E/S (toutes les opérations d'E/S se font sur le même périphérique alloué par ordre FIFO).

Processus :	P1	P2	P3
Temps CPU :	4	6	3
Durée E/S :	7	3	2
Temps CPU :	3	2	3

L'instant d'arrivée: 0 1 3

- 1) Ces processus sont traités dans l'ordre d'arrivée, selon l'algorithme Best-Fit, quelles sont les zones qu'occupe chacun des processus ?
- 2) En utilisant l'algorithme Round Robin avec un quantum de temps égal à 2 unités, donnez le diagramme de Gant (chronogramme d'exécution des processus) et calculez le temps d'attente moyen et le temps de séjour moyen.

Solution

1) P1 dans la partition 13Ko ; P2 dans la partition 15Ko ; P3 dans la partition 25Ko. (0,25*3)

2) Diagramme de Gant

	E/S						x	x	x	x	x	x	x	x						
P1	Actif	x	x		x	x								x	x			x		
	Prêt			x	x									x	x					

0,5pts 0,5pts 0,5pts

	E/S													x	x	x			
P2	Actif			x	x					x	x		x	x				x	x
	Prêt		x		x	x	x	x		x						x			

0,5pts 0,5pts 0,5pts

	E/S													x	x	x			
P3	Actif					x	x			x				x	x	x		x	
	Prêt			x	x	x			x	x						x			

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

A chaque processus : 0,5pts 0,5pts 0,5pts

$$\begin{aligned}
 T_{attMoy} &= (TattP1 + TattP2 + TattP3)/3 \\
 &= [(18-0-7-7) + (21-1-3-8) + (19-3-2-6)]/3 = (4+9+8)/3 \\
 &= 7
 \end{aligned}
 \quad \begin{matrix} (0,25*3) \\ 0,25pts \end{matrix}$$

$$\begin{aligned}
 T_{sejMoy} &= (TsejP1 + TsejP2 + TsejP3)/3 \\
 &= [(18-0) + (21-1) + (19-3)]/3 = (18 + 20 + 16)/3 \\
 &= 18
 \end{aligned}
 \quad \begin{matrix} (0,25*3) \\ 0,25pts \end{matrix}$$