

 <b>esprit</b> <small>Se former autrement</small> <small>HONORIS UNITED UNIVERSITIES</small>	<p align="center"><b>Examen</b></p> <p><b>Semestre 1</b> <b>Session Principale</b></p>
<p><b>Module :</b> Système d'exploitation avancé 1  <b>Enseignant(s) :</b> UP Système  <b>Classe(s) :</b> 4 année</p>	<p><b>Documents non autorisés</b>  <b>Nombre de pages :</b> 2  <b>Date :</b> 13/11/2023 <b>Heure</b> 11h00 <b>Durée :</b> 01h30</p>

### Exercice 1 : ordonnancement des processus (6 points)

A/ Supposons qu'un système d'exploitation gère cinq processus (A, B, C, D, et E) qui arrivent à différents moments et doivent être exécutés sur un seul processeur. Chaque processus a une durée d'exécution différente, et le système utilise l'ordonnancement préemptif en utilisant la priorité comme critère.

Les caractéristiques des processus sont les suivantes :

Processus	Temps d'arrivée	Temps d'exécution	Priorité
A	0	6	3
B	2	4	2
C	4	8	1
D	5	2	3
E	6	3	2

### Remarque : priorité i est supérieure par rapport à la priorité i-1

À chaque instant, le système choisit le processus avec la priorité la plus élevée. En cas d'égalité de priorité, le processus qui arrive en premier est choisi. Si un processus termine son exécution ou est préempté, le processus avec la priorité la plus élevée est choisi.

1. Quel sera l'ordre d'exécution des processus ? Fournissez une séquence d'exécution avec l'heure d'arrivée, la priorité, le temps d'exécution restant, et le processus en cours.
2. Quelle est la durée totale nécessaire pour exécuter les trois processus dans cet ordre ?
3. Donnez le temps moyen de séjour (temps moyen de rotation).

B/ Supposons que vous ayez maintenant un système d'exploitation utilisant l'algorithme d'ordonnancement Round Robin. L'intervalle de temps de chaque quantum (temps de rafraîchissement) est de 3 unités de temps.

1. Appliquez l'algorithme Round Robin pour ordonner l'exécution de ces processus. Fournissez une séquence d'exécution montrant l'heure d'arrivée, le temps d'exécution restant pour chaque processus à chaque quantum, et le processus en cours.
2. Quel est le temps d'attente moyen des processus dans cet ordre d'exécution ?

**Exercice 2 : (7 points)**

Écrire un programme **P** qui crée deux fils **F1** et **F2**, tous les processus doivent afficher le **PID** des trois processus avant de se terminer proprement.

**Exercice 3 : (7 points)**

Ecrire un programme qui crée 2 processus l'un faisant la commande **ls -l**, l'autre **ps -l**. Le père devra attendre la fin de ses deux fils et afficher le **PID** du premier processus terminé.

**Bonne chance**