



Filière : 1 LInfo	Matière : Système exploitation 2		Enseignant : Sana BENZARTI
Date : 13/03/2024			Documents autorisés : Non Calculatrice autorisée : oui
Durée de l'examen : 1h	Régime d'évaluation : Mixte / CC		Nombre de pages : 2

La clarté et la propreté de la feuille de l'examen sont prises en compte

Exercice 1 : QCM: Vous pouvez trouver plusieurs réponses exactes pour une même question

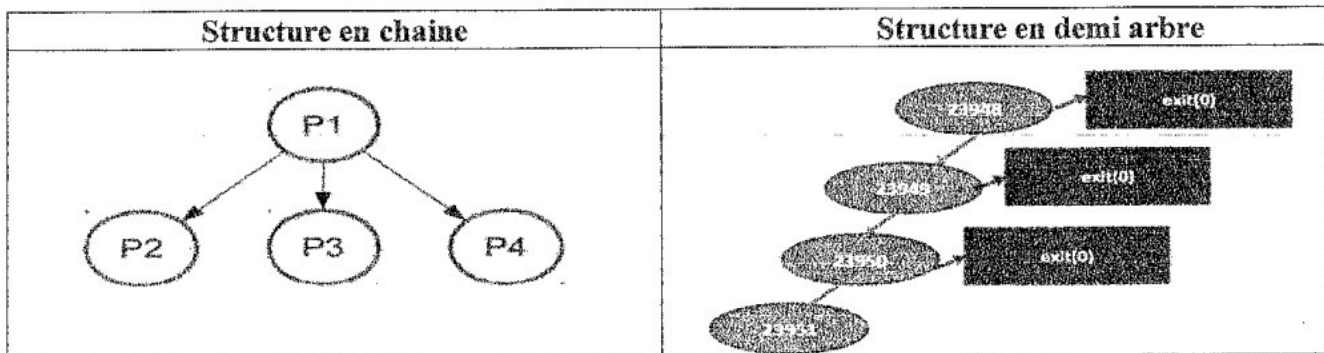
1- Le PCB contient : a- L'état du processus b- La valeur du compteur ordinal	6- Le PPID est renvoyé uniquement par : a- Le processus père b- Le processus fils c- Le processus parent
2- Intel Core i5-11600K, il s'agit d'un processeur i5 de la génération : a- 1 b- 11 c- 116	7- La fonction wait() permet de : a- Attendre un processus fils particulier b- Attendre n'importe quel processus fils c- Renvoyer le PID du processus fils chez le père
3- AMD Ryzen 9 5900X, il s'agit d'un processeur AMD Ryzen de la génération : a- 9 b- 5 c- 59	8- Repose sur des pièces mobiles avec un bras mécanique, il s'agit d'un disque : a- SSD b- HDD
4- Un processus fils devient zombie si : a- Toutes ses ressources sont indisponibles b- Son père meurt avant lui c- Il meurt avant son père	9- Utilise la mémoire flash NAND pour lire et écrire les données, c'est un disque : a- SSD b- HDD
5- Les processus orphelins sont adoptés par le processus : a- Père b- Init c- Systemd	10- La fonction fork() retourne : a- La valeur 0 chez le processus père b- Le PID du fils chez le processus père c- La valeur -1 chez le processus père

Exercice 2 : Les fonctions fork(), wait() et exit()

A- On considère les codes 1 et 2. Pour chaque code, mentionner le nombre de processus générés avec le nombre d'affichage du mot "hello".

Code 1	Code 2
<pre>int main() { if (fork()) fork(); printf("hello!\n"); exit(0); }</pre>	<pre>int main() { if (fork() != 0) { if (fork()) { printf("hello!\n"); } } }</pre>

B- Ecrire un code en c qui crée 3 processus fils chacun d'eux affiche son PID et le PPID en respectant le structure en chaine et celle en demi arbre.



Exercice 3 : Ordonnancement des processus

Soit le tableau suivant :

Processus	Date d'arrivée	Temps de traitement
P ₁	0	3
P ₂	2	6
P ₃	4	4
P ₄	6	5
P ₅	8	2

En utilisant les algorithmes d'ordonnancement SJF (sans préemption) et Round Robin (avec préemption, Quantum = 4 et file d'attente circulaire) :

- Donner le diagramme de Gantt
- Calculer sous forme de tableau et pour chaque algorithme : le temps de rotation, le temps d'attente, la moyenne du temps de rotation, la moyenne du temps d'attente et le débit (mettre les détails du calculs).

Bon travail