

**1) Un processus est :**

- a) Un fichier dont je peux lire le contenu avec 'cat'
- b) Un programme en cours d'exécution **x**
- c) Un composant électronique inclus dans le processeur

**2) Qu'est-ce qu'un PID ?**

- a) un identifiant donné à une entrée de la table de partition
- b) un identifiant donné à un compte utilisateur
- c) un identifiant donné à un processus **x**

**3) Le 'PID' représente :**

- a) Un numéro de groupe
- b) Une adresse de la machine
- c) Un numéro de processus **x**
- d) Un numéro unique **x**

**4) 'init' est :**

- a) processus placé en premier dans la commande 'ps'
- b) La commande qui permet de réinitialiser son profil
- c) Le premier processus de la parenté **x**

**5) Un processus, autre que le processus init :**

- a) Peut ne pas avoir de processus parent
- b) Peut avoir deux processus parents
- c) N'a qu'un seul processus parent **x**

**6) Un utilisateur peut créer un processus**

- a) En tapant la commande 'proc' suivi de l'identificateur du processus
- b) En tapant une commande dans le shell **x**
- c) En cliquant sur l'icône d'une application **x**

**7) Le Kernel est :**

- a) Une partie du système d'exploitation se trouvant dans le boot
- b) Une partie du système d'exploitation résidente en mémoire centrale **x**
- c) Une partie du système d'exploitation résidente en mémoire secondaire

**8) Un Process Control Block (PCB) est:**

- a) La pile d'exécution d'un processus
- b) Une zone mémoire qui définit le contexte d'un processus **x**
- c) Le PID d'un processus

**9) Le PCB d'un processus contient :**

- a) Le code exécutable du processus
- b) l'état d'un processus **x**
- c) Les registres du processeur

**10) Citer quatre attributs parmi ceux qui constituent le PCB.**

**PID, PPID, Etat du processeur, Priorité, Compteur Ordinal, Fichiers ouverts, pointeurs (seg de code, seg de données, seg de pile..), temps d'exécution...**

**11) Le shell :**

- a) Est exécuté dans le noyau du système d'exploitation Unix
- b) Est un programme structuré comme une boucle infinie dont on sort par 'exit' **x**
- c) Exécute les commandes en créant dans la plus part des cas un nouveau processus **x**

**12) Les interpréteurs de commandes font-ils parties du système d'exploitation?**

- a) oui **x**
- b) non

**13) Que se passe-t-il lorsqu'un processus devient orphelin?**

Le processus orphelin sera adopté par le processus INIT (le premier processus créé au démarrage de la machine) afin de récupérer son statut et demander sa destruction et récupération des ressources qu'il détenait (comme l'espace mémoire occupé par le code et par son PCB ...).

**14) Quand est-ce un processus passe à l'état Zombie ?**

Un processus est à l'état zombie lorsque son père n'a pas atteint l'exécution de la fonction « wait » ou « waitpid » lui permettant de récupérer son statut. Il s'agit du processus fils qui termine alors que son père n'a pas encore exécuté « wait » ou « waitpid » dans son code. Ainsi le fils reste dans cet état jusqu'à ce que son père exécute « wait » ou « waitpid » ensuite il passe à l'état terminé.

**15) Où se trouve l'information sur les processus dans le système d'exploitation?**

- a) Dans l'ordonnanceur de processus
- b) Dans la table des processus **x**
- c) Dans le shell

**16) Un thread :**

- a) S'exécute sur plusieurs coeurs, ou processeur, simultanément
- b) Est composé de plusieurs processus
- c) Partage le même espace de mémoire avec tous les threads composant son processus **x**

**17) Pour lancer en parallèle plusieurs traitements d'une même application, vous avez le choix entre les appels système `fork()` et `pthread_create()`. Laquelle des deux possibilités choisir? Pourquoi?**

L'application peut être composée de plusieurs processus ou plusieurs threads ; la différence est qu'avec les processus chacun occupe un espace propre à lui en mémoire centrale alors que les threads se partagent l'espace de travail entre eux en mémoire centrale. De plus le changement de contexte est plus court en utilisant les threads. Donc il est préférable d'utiliser `pthread_create()`.

**18) Qu'est-ce qu'un changement de contexte?**

- a) Un appel système
- b) Le changement du processus en exécution dans le processeur **x**
- c) Le remplacement de l'image mémoire d'un processus

**19) A l'exécution de la fonction `fork()`, quelle est la différence entre le processus père et fils ?**

La différence est leurs pid : à l'exécution de `fork()` par le père, le pid du fils sera le pid de son père + 1.

**20) Quelle est la raison pour laquelle le processus Init adopte les processus orphelins ?**

Le processus Init adopte les processus orphelins pour récupérer leurs états de terminaisons et les détruire afin d'éviter leurs occupations mémoire inutiles puisqu'ils ont terminés.