



Math93.com

TD NSI - Processus

Exercice 1

Un constructeur automobile utilise des ordinateurs pour la conception de ses véhicules. Ceux-ci sont munis d'un système d'exploitation ainsi que de nombreuses applications parmi lesquelles on peut citer :

- un logiciel de traitement de texte;
- un tableur;
- un logiciel de Conception Assistée par Ordinateur (CAO);
- un système de gestion de base de donnée (SGBD).

Chaque ordinateur est équipé des périphériques classiques : clavier, souris, écran et est relié à une imprimante réseau. Un ingénieur travaille sur son ordinateur et utilise les quatre applications citées ci-dessus. Pendant l'exécution de ces applications, des processus mobilisent des données et sont en attente d'autres données mobilisées par d'autres processus. On donne ci-dessous un tableau indiquant à un instant précis l'état des processus en cours d'exécution et dans lequel D1, D2, D3, D4 et D5 sont des données. La lettre M signifie que la donnée est mobilisée par l'application; la lettre A signifie que l'application est en attente de cette donnée. Lecture du tableau : le logiciel de traitement de texte mobilise (M) la donnée D1 et est en attente (A) de la donnée D2.

	D1	D2	D3	D4	D5
Traitement de texte	M	A	-	-	-
Tableur	A	-	-	-	M
SGBD	-	M	A	A	-
CAO	-	-	A	M	A

Montrer que les applications s'attendent mutuellement. Comment s'appelle cette situation?

Exercice 2

La commande UNIX *ps* présente un cliché instantané des processus en cours d'exécution.

Avec l'option *-eo pid, ppid, stat, command*, cette commande affiche dans l'ordre l'identifiant du processus *PID* (*process identifier*), le *PPID* (parent process identifier), l'état *STAT* et le nom de la commande à l'origine du processus.

Les valeurs du champ *STAT* indique l'état des processus :

R : processus en cours d'exécution

S : processus endormi

Sur un ordinateur, on exécute la commande *ps -eo pid, ppid, stat, command* et on obtient un affichage dont on donne ci-dessous un extrait.

```
$ ps -eo pid,ppid,stat,command
```

```
PID PPID STAT COMMAND
1 0 Ss /sbin/init
.... ..
1912 1908 Ss Bash
2014 1912 Ss Bash
1920 1747 Sl Gedit
2013 1912 Ss Bash
2091 1593 Sl /usr/lib/firefox/firefox
5437 1912 Sl python programme1.py
5440 2013 R python programme2.py
5450 1912 R+ ps -eo pid,ppid,stat,command
```

À l'aide de cet affichage, répondre aux questions ci-dessous.

1. Quel est le nom de la première commande exécutée par le système d'exploitation lors du démarrage?
2. Quels sont les identifiant des processus actifs sur cet ordinateur au moment de l'appel de la commande *ps*? Justifier la réponse.

3. Depuis quelle application a-t-on exécuté la commande *ps*? Donner les autres commandes qui ont été exécutées à partir de cette application.
4. Expliquer l'ordre dans lequel les deux commandes `python programme1.py` et `python programme2.py` ont été exécutées.
5. Peut-on prédire que l'une des deux commandes `python programme1.py` et `python programme2.py` finira avant l'autre?

Exercice 3

1. Les états possibles d'un processus sont : prêt, élu, terminé et bloqué.
 - (a) Expliquer à quoi correspond l'état élu.
 - (b) Proposer un schéma illustrant les passages entre les différents états.
2. On suppose que quatre processus C_1 , C_2 , C_3 et C_4 sont créés sur un ordinateur, et qu'aucun autre processus n'est lancé sur celui-ci, ni préalablement ni pendant l'exécution des quatre processus. L'ordonnanceur, pour exécuter les différents processus prêts, les place dans une structure de données de type file. Un processus prêt est enfilé et un processus élu est défilé.
 - (a) Parmi les propositions suivantes, recopier celle qui décrit le fonctionnement des entrées/sorties dans une file :
 - i Premier entré, dernier sorti
 - ii Premier entré, premier sorti
 - iii Dernier entré, premier sorti
 - (b) On suppose que les quatre processus arrivent dans la file et y sont placés dans l'ordre C_1 , C_2 , C_3 et C_4 . Les temps d'exécution totaux de C_1 , C_2 , C_3 et C_4 sont respectivement 100 ms, 150 ms, 80 ms et 60 ms.
 Après 40 ms d'exécution, le processus C_1 demande une opération d'écriture disque, opération qui dure 200 ms. Pendant cette opération d'écriture, le processus C_1 passe à l'état bloqué.
 Après 20 ms d'exécution, le processus C_3 demande une opération d'écriture disque, opération qui dure 10 ms. Pendant cette opération d'écriture, le processus C_3 passe à l'état bloqué.
 Sur la frise chronologique donnée en annexe (à rendre avec la copie), les états du processus C_2 sont donnés. Compléter la frise avec les états des processus C_1 , C_3 et C_4 .
3. . On trouvera ci- dessous deux programmes rédigés en pseudo-code
 Verrouiller un fichier signifie que le programme demande un accès exclusif au fichier et l'obtient si le fichier est disponible.

<u>Programme 1</u>	<u>Programme 2</u>
Verrouiller fichier_1	Verrouiller fichier_2
Calculs sur fichier_1	Verrouiller fichier_1
Verrouiller fichier_2	Calculs sur fichier_1
Calculs sur fichier_1	Calculs sur fichier_2
Calculs sur fichier_2	Déverrouiller fichier_1
Calculs sur fichier_1	Déverrouiller fichier_2
Déverrouiller fichier_2	
Déverrouiller fichier_1	

- (a) En supposant que les processus correspondant à ces programmes s'exécutent simultanément (exécution concurrente), expliquer le problème qui peut être rencontré.
- (b) Proposer une modification du programme 2 permettant d'éviter ce problème.

