

## Les processus et les variables d'environnement

### Les processus :

Dans le langage c sous linux il est possible qu'un processus crée d'autres processus en utilisant l'instruction suivante :

**Int etat=fork() :**

La variable état va être égale à :

- **0** : dans le processus fils (processus créé).
- **Pid du processus fils** : dans le processus père.
- **-1** : si le processus fils n'a pas été créé.

Afin d'avoir des informations sur le contexte dans lequel un processus se trouve on peut utiliser les fonctions suivantes :

**int getpid() // qui indique l'identifiant du processus**  
**int getppid() // qui indique l'identifiant du processus père**  
**int getgid() // qui indique l'identifiant du groupe**  
**int getuid() // qui indique l'identifiant de l'utilisateur**

Pour terminer un processus, on utilise la fonction

**void exit(int status)**

Pour attendre la fin d'exécution d'un processus, on utilise la fonction :

**pid\_t wait(int \*status);**  
**pid\_t waitpid(pid\_t pid, int \*status, int options);**

Tag	Description
< -1	Attend la fin d'un fils processus qui a un groupe ID égale à la valeur absolue du <i>pid</i> passé en paramètre.
-1	Attend la fin d'un processus fils.
0	Attend la fin d'un fils qui a le même group ID que le processus appelant.
> 0	Attend le processus fils qui porte le <i>pid</i> passé en paramètre.

On peut récupérer l'état status en utilisant la fonction suivante :

```
WEXITSTATUS(status) ;
```

### **Les variables d'environnement :**

Les variables d'environnement sont des variables dynamiques utilisées généralement par les différents processus d'un système d'exploitation pour communiquer des informations entre programmes qui ne se trouvent pas sur la même ligne hiérarchique.

Vous pouvez initialiser des variables d'environnement via le terminal comme suit :

```
Nom_variable=valeur
```

Si le nom de la variable existe déjà dans la liste des variables d'environnements, la variable va prendre la valeur indiquée. Par contre, si la variable ne figure pas dans la liste des variables d'environnement vous devez ajouter la commande suivante :

```
Export Nom_variable
```

Il est possible aussi de récupérer les variables d'environnement au niveau d'un programme c en déclarant la fonction main comme suit :

```
Int main(int arg, char* argument[], char* var_environnement[]) ;
```

On peut aussi récupérer les variables d'environnement dans un programme en utilisant la variable globale environ, qui doit être déclarée au niveau du programme comme suit :

```
Extern char** environ ;
```

Pour récupérer la valeur d'une variable d'environnement bien précise au niveau d'un programme c, on peut utiliser la fonction getenv() comme suit :

```
Char * variable=getenv(nom_variable) ;
```

Enfin, on peut aussi modifier la valeur d'une variable d'environnement en utilisant la fonction putenv() comme suit :

```
Int i=Putenv(«expression du genre nom_variable=valeur ») ;
```