



Nom :

Prénom :

Groupe :

Interrogation TP(Systèmes d'Exploitation 1)

Date : 08/05/2024

Durée: 45 minutes - Documentation non autorisée

Exercice 1 : (7 Pts) (20 minutes)

0.5x10 = 5 Pts

Q1) Qui suis-je ?

Je suis une commande qui permet de revenir au répertoire /home/utilisateur	cd ou bien cd ~
Je suis une commande qui affiche le contenu du répertoire	ls
Je suis une commande qui permet de remonter au répertoire parent	cd ..
Je suis une commande qui crée le répertoire	mkdir
Je suis une commande qui Modifie les permissions d'accès à un fichier ou à un répertoire.	chmod
Je suis une commande qui affiche les processus en cours d'exécution sous la forme d'un arbre.	pstree
Je suis un appel système qui permet de créer un processus	fork()
Je suis un appel système qui affiche le PID du processus père.	getppid()
Je suis une commande qui envoie un signal au processus.	kill
Je suis une commande qui affiche les processus en cours d'exécution sous la forme d'un arbre.	pstree

Q2) Quel est le rôle de l'instruction : wait(NULL) ;

Le processus père se bloque jusqu'à la terminaison d'un de ses processus fils.

1

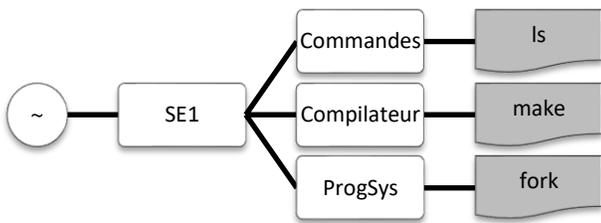
Q3) Quel est le rôle de l'instruction : while (wait(NULL) >= 0);

Le processus père se bloque jusqu'à la terminaison de tous ses processus fils.

1

Exercice 2 : (5 Pts) (15 minutes)

Q1) Donnez les instructions Unix qui permettent de créer l'arbre des dossiers et des fichiers suivants :



Réponse :

```
mkdir SE1
cd SE1
mkdir Commandes
mkdir Compilateur
mkdir ProgSys
touch /SE1/Commandes/ls
touch /SE1/Compilateur/make
touch /SE1/ProgSys/frok
```

Q2) Donnez les instructions Unix qui permettent de se déplacer vers le dossier Commandes et d'afficher son contenu.

```
cd /SE1/Commandes
ls
```

Q3) Modifier les droits d'accès du fichier make pour qu'ils soient : 744

```
chmod 744 /SE1/Compilateur/make
```

Exercice 3 : (3 Pts) (10 minutes)

Écrire un programme C qui permet de générer un nombre N de processus. Le nombre N est préalablement saisi par l'utilisateur.

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>

int main ()
{
    int i, n ;
    printf ( " Entrer la valeur de n ?\n " ) ;
    scanf ("%d" ,&n ) ;
    for ( i = 0 ; i < n ; i ++ )
        if ( fork () == 0 ) exit(0) ; // ou bien instruction if ( fork () > 0 ) exit(0);
    return 0 ;
}
```