Partie 3 : Signaux

Exercice 1 :

Le signal SIGCHLD est un signal qui est automatiquement envoyé par le fils à son père lorsque le fils se termine (par un exit, un return, ou autre).

Ajoutez une fonction et **le code nécessaire** afin que le père n'attende jamais son fils de façon bloquante et que le fils ne devienne pas zombie.

```
/*0*/
int main(int argc, char *argv[])
{
        /*1*/
        if (!fork())
        {
                /*2*/
                for (int i = 0 ; i <10 ; i++) ; //simule un petit calcul
                /*3*/
                exit(1) ;
                /*4*/
        }
        /*5*/
        while(1) ; //Simule un calcul infini
        /*6*/
}
```

Exercice 2 :

Lorsqu'un processus tente d'écrire dans un pipe rompu (fermé en lecture), le signal SIGPIPE est envoyé au processus. Le traitement par défaut associé à ce signal est la terminaison.

Expliquez ce qu'il faudrait faire pour que le processus **capte le signal** et **affiche** un message d'erreur qui spécifie lequel des pipes est rompu (son descripteur de fichier) avant de se terminer.

Exercice 3 :

Considérez le programme suivant :

```
#include <unistd.h>
#include <sys/types.h>
#include <sys/wait.h>
#include<stdio.h>
#include <signal.h>
#define N 5
/*0*/
int main( )
{       pid_t pid[N];
        int i ;
        /*1*/
        for ( i=0; i<N;i++)
            if ((pid[i]=fork())==0)
            {   /*2*/
                while(1)
                    printf("ici fils %d ", i);
            }

         /*3*/

}
```

On veut que le processus père utilise les signaux SIGSTOP et SIGCONT pour suspendre (bloquer) et reprendre (débloquer) l'exécution de ses processus fils. Au départ tous les processus fils créés doivent se mettre en pause. Le processus père répète continuellement le traitement suivant en commençant par le premier fils : il envoie le signal SIGCONT à un fils puis s'endort pendant 1 seconde. À son réveil, il envoie SIGSTOP au même fils et SIGCONT au fils suivant (le fils suivant du dernier est le premier).

Lorsqu'un processus fils reçoit le signal SIGCONT, il affiche le message indiquant qu'il a capturé le signal SIGCONT avant de poursuive son exécution.

Compléter le code précédent de manière à réaliser ce traitement.