

```

/* Les signaux sous UNIX */
/* TDM2 Exercice 1 */
#include <sys/types.h>
#include <stdlib.h>
#include <unistd.h>
#include <signal.h>
#include <stdio.h>

typedef enum {FAUX=0,VRAI=1} boolean;
typedef void ( *PtrFct) (int);

void traiterSIGUSR1( int );

int main ( )
{
    PtrFct retFct ;
    retFct=signal( SIGUSR1, traiterSIGUSR1 );
    if (retFct == SIG_ERR)
        { perror("echec signal"); exit(1); }
    while (VRAI)
    {
        sleep(5);
    }
}

void traiterSIGUSR1( int sig )
{
    switch ( sig )
    {
        case SIGUSR1 :
            printf("PID= %d\n", getpid() );
            printf("n° du signal reçu = %d\n", sig );
            break;
        default :
            printf("\n Erreur système !!!!\n");
    }
    exit(2);
}

/* Les signaux sous UNIX */
/* TDM2 Exercice 2 */
#include <sys/types.h>
#include <stdlib.h>
#include <unistd.h>
#include <signal.h>
#include <stdio.h>

int main(int nbParam, char * tabParam[])
{
    int err;
    pid_t pid;
    switch (nbParam)
    {
        case 2 :
            sscanf(tabParam[1], "%d", &pid);
            printf("envoi du signal SIGUSR1 au processus= %d\n", pid);
            err=kill( pid, SIGUSR1 );
            if (err == -1)
                { perror("echec kill");exit(1); }
            break;
        default :
            printf("***nombre de parametres incorrect!!!\n"); exit(1);
    }
}

/* Les signaux sous UNIX */
/* TDM2 Exercice 3 */
#include <sys/types.h>
#include <stdlib.h>
#include <unistd.h>

```

```
#include <signal.h>
#include <stdio.h>
#include <setjmp.h>

#define TIMEOUT1 1

typedef void(* PtrFct)(int);

void timeout( int );

jmp_buf ptRep; /* doit etre declare en variable globale */

int main ( )
{
    PtrFct retFct;
    int ret;
    int nbEssai=0;
    unsigned duree=10; /* delai d'attente 10 sec */
    char mess[257];
    retFct=signal( SIGALRM, timeout );
    if (retFct == SIG_ERR) { perror("echec signal"); exit(1); }
    ret=setjmp(ptRep);
    if (ret == TIMEOUT1 )
    {
        if (nbEssai > 2 ) { printf("echec saisie!!!\n"); exit(2); }
    }
    nbEssai++;
    printf("entrer votre message:\n");
    alarm(duree);
    fgets(mess,256,stdin);
    alarm(0);
    printf("saisie OK !!!\n");
}

void timeout( int sig )
{
    switch ( sig )
    {
        case SIGALRM :
            printf("n° du signal reçu = %d\n", sig );
            sigelse(SIGALRM); /* demasquer le signal SIGALRM */
            longjmp(ptRep,TIMEOUT1);
        default :
            printf("\n Erreur système !!!!\n");
            exit(3);
    }
}
```