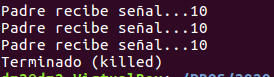
**ACTIVIDAD 10**

**Enunciado**

Realiza un programa C en donde un hijo envíe 3 señales SIGUSR1 a su padre y después envíe una señal SIGKILL para que el proceso padre termine.

**Resultado esperado:**



**Solución:**

// actividad10.c

// Joseba Martinez

#include <stdio.h>

#include <stdlib.h>

#include <sys/types.h>

#include <sys/stat.h>

#include <fcntl.h>

#include <unistd.h>

#include <signal.h>

pid\_t pid\_hijo,pid\_padre;

void manejador (int signal){

printf("Padre recibe señal...%d\n",signal);

}

int main (void){

pid\_padre = getpid();

pid\_hijo = fork();

switch (pid\_hijo)

{

case -1:

printf("No se ha podido crear el proceso hijo\n");

exit(-1);

break;

case 0:

kill (pid\_padre,SIGUSR1);

sleep(1);

kill (pid\_padre,SIGUSR1);

sleep(1);

kill (pid\_padre,SIGUSR1);

sleep(1);

kill(pid\_padre,SIGKILL);

break;

default:

signal(SIGUSR1,manejador);

while (1){

sleep(2);

}

break;

}

return(0);

}

**Resolución de problemas:**

**CRITERIOS DE CORRECCIÓN**

1. Nivel de ajuste a la nomenclatura de objetos definida en el enunciado de la actividad y nivel de ajuste al almacenamiento de la información requerida en las subcarpetas definidas para la actividad y unidad didáctica correspondientes (15%).
2. Correcto funcionamiento de los programas desarrollados en C (35%).
3. Ajuste a los resultados esperados (35%).
4. Claridad del código C (15%).
5. Explicación detallada de cómo se han resuelto problemas que hayan podido surgir en la realización de la actividad (hasta un 15% adicional sobre la nota obtenida).
6. Avisos o errores de compilación (-15%).
7. Nivel de comentarios en el código fuente (-15%).