

```
#include <unistd.h>
#include <stdio.h>
#include <stdlib.h>
#include <signal.h>
#include <sys/wait.h>
#include <sys/types.h>

#define BUFFSIZE 512

int fib(int n)
{
    if(n <= 1)
    {
        return n;
    }

    return (fib(n-1) + fib(n-2));
}

int fatt(int n)
{
    if(n <= 1)
    {
        return 1;
    }

    return n * fatt(n-1);
}

void handler_f1(int signo)
{
    signal(SIGINT, SIG_IGN);
    printf("[FIGLIO1][HANDLER] ricevuto segnale %d fib 10: %d\n", signo, fib(10));
    signal(SIGINT, handler_f1);
}

void handler_f2(int signo)
{
    signal(SIGINT, SIG_IGN);
    char answer[BUFFSIZE];
    printf("[FIGLIO2][HANDLER] ricevuto segnale %d: continuare(c) o terminare(q)? : \n",
signo);
    scanf("%s", answer);
    if(*answer == 'c')
    {
        signal(SIGINT, handler_f2);
    }
    else
    {
        printf("terminazione programma\n");
        exit(1);
    }
}
```

```
int main(void)
{
    signal(SIGINT, SIG_IGN);
    pid_t f1, f2;

    f1 = fork();

    if(f1 == 0)
    {
        //FIGLIO1
        signal(SIGINT, handler_f1);
        for(int i = 0; i < 30; i++)
        {
            printf("[FIGLIO1] fib %d: %d\n", i, fib(i));
            sleep(2);
        }
        exit(EXIT_SUCCESS);
    }
    else
    {
        //PADRE
        f2 = fork();
        if(f2 == 0)
        {
            //FIGLIO2
            signal(SIGINT, handler_f2);
            for(int i = 0; i < 20; i++)
            {
                printf("[FIGLIO2] fatt %d: %d\n", i, fatt(i));
                sleep(2);
            }
            exit(EXIT_SUCCESS);
        }

        //PADRE
        wait(NULL);
        wait(NULL);
        printf("figlio %d e figlio %d somma: %d terminati", f1, f2, (f1+f2));
    }
    return 0;
}
```