Nome: Luca Boffa

Matricola:0215106052

#include <stdio.h>

#include <stdlib.h>

#include <sys/wait.h>

#include <unistd.h> //libreria aggiunta

int int main()

{

int pid1 , pid2 , status;

signal(SIGINT,SIG\_IGN);

pid1 = fork();

if ( pid1 < 0)

fprintf(stderr , "Errore");

else if (pid1 == 0)

{

printf(" pid figlio 1 %d \n", getpid()); // aggiunto \n e rimmosso il pid come argomento della funzione

for (int i = 0 ; i < 50 ; i++)

{

sleep(1); //aggiunto sleep per avere il tempo ad inviare SIGINT

printf(" %d \n" , i+1);

}

exit(0); //aggiunta exit

}

else if (pid1 > 0)

pid2 = fork();

if (pid2 < 0)

fprintf(stderr, "Errore");

else if (pid2 == 0)

{

printf(" pid padre %d \n" , getppid()); //avevo scritto getpipid e aggiunto il pid come argomento

execlp("ls", "ls", NULL ); //aggiunte virgolette e modificato il gli argomenti della funzione

exit(0); //aggiunto exit

}

else

{

printf(" Terminazione figlio 1 %d \n ", waitpid(pid1 ,&status , 0)); //aggiunto 0

printf(" Terminazione figlio 2 %d \n ", waitpid(pid2 , &status , 0)); //aggiunto 0

}

return 0; //aggiunto return 0

}

OUTPUT:

pid figlio 1 1769

pid padre 1678

doc1.c esercizi a.out newshell.c SistemiOperativi

1

2

3

….

….

50

Terminazione figlio 1 con pid 1769

Terminazione figlio 2 con pid 1770