**Autocorrezione Esercizio Fork()**

#include<sys/types.h>

#include<sys/wait.h>

#include<unistd.h>

#include<stdio.h>

#include<stdlib.h>

#include<signal.h>

int main()

{

/\*

Creazione del primo figlio

\*/

int pid1= fork();

int status, i=0, numero=0;

signal(SIGINT, SIG\_IGN); //Ignora il segnale immesso da input

if(pid1<0)

{

printf("Errore nella fork\n");

}

else if(pid1==0)

{

signal(SIGINT, SIG\_IGN);

printf("Il mio pid e' (pid1): %d \n", getpid());

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

{

numero= numero+1;

printf("Il numero e': %d \n", numero);

sleep(1);

}

exit(1);

}

else if(pid1>0)

{

/\*

Creazione del secondo figlio

\*/

int pid2 = fork();

if(pid2<0)

{

printf("Errore nella fork\n");

}

else if(pid2==0)

{

signal(SIGINT, SIG\_IGN);

printf("Il pid di mio padre e': %d \n", getppid());

execlp("/bin/ls", " ", NULL);

printf(“Il pid2 ha eseguito il comando /bin/ls \n”);

exit(1);

}

else if(pid2>0)

{

signal(SIGINT, SIG\_IGN);

wait(&status); //Aspetta la terminazione del Primo figlio

wait(&status); //Aspetta la terminazione del Secondo figlio

printf("Il mio pid e': %d \n", pid1);

printf("Il mio pid e': %d \n", pid2);

printf(“Il pid1 e il pid2 sono terminati \n”);

exit(1);

}

}

return 0;

}

**Descrizione Modifiche**

Per il corretto funzionamento del codice sono state apportate le seguenti modifiche:

* Ho aggiungo la libreria #include<stdio.h>.
* Ho spostato la dichiarazione int pid2=fork() all’interno della condizione if(pid1>0).
* Ho corretto qualche errore di sintassi commesso per distrazione.