Esercitazione Pipe

File system di Linux: inode e system call

Esercitazione sulle pipe

1) Il padre legge un intero (o un array) lo passa al figlio che fa il quadrato del numero, lo restituisce al padre che lo stampa su schermo

2) Il padre legge un vettore da tastiera, quindi crea 2 figli F1 e F2, manda i numeri multipli di 2 al figlio F1, i multipli di 3 a F2. F1 calcola la media dei multipli di 2, mentre F2 calcola il massimo dei multipli di 3. Ritornano i risultati al padre.

Per semplicita si suppone che non esista lo 0 fra i numeri letti.

P:

Crea i figli

legge un vettore da tastiera

manda i multipli di 2 a F1

manda i multipli di 3 a F2

Padre aspetta i risultati..

Pipe necessarie:

PF1, PF2. -> manda vettore ai figli

padre scrive. (close (PF1 in lettura), figli leggono. (close (PF1 in scrittura)

F1P, F2P1 -> MANDA RISULTATI AL PADRE

Figli scrivono, padre legge

pipe(PF1)

pid=fork()

Padre e figlio condividono la pipe

pid=fork()

pipe(PF1)

Padre e figlio hanno due versioni diverse della pipe

3) Svolgere l'esercizio 1 per 3 figli (considerare multipli di 2, 3 e 4).

4) Il padre legge un vettore da tastiera, quindi crea 2 figli Fpos e Fneg.

Il padre passa i numeri positivi a Fpos e i negativi a Fneg. Fpos calcola la media dei numeri positivi e la ritorna al padre che la stampa. Fneg crea un nipote Fnip, gli passa i numeri positivi e Fnip calcola il minimo dei numeri negativi e lo ritorna a Fneg che lo ritorna al padre che lo stampa.

Il padre passa i numeri positivi a Fpos e i negativi a Fneg. Fpos calcola la media dei numeri positivi e la ritorna al padre che la stampa.

Fneg crea un nipote Fnip, gli passa i numeri negativi e Fnip calcola il minimo dei numeri negativi e lo ritorna a Fneg che lo ritorna al padre che lo stampa.

// codice FNEG

pipe(FNEG\_NIP)

pipe(NIP\_FNEG)

pid2 =fork();

if (pid2==0){

//codice nipote

}

else{

codice FNEG

}

VFS Virtual file system

ext4

inode

Blocco di dati su disco = 1k

file di 100 k= 100 blocchi

Indirizzamento indiretto