

```
gcc -o lab3 lab3.c
./lab3 6
```

Οι διεργασίες επικοινωνούν και συγχρονίζονται μέσω διοχετεύσεων (pipes).

- Λειτουργούν ως σωληνώσεις που επιτρέπουν σε δύο διεργασίες να επικοινωνούν
- Οι συνηθισμένες διοχετεύσεις δεν είναι προσβάσιμες έξω από την διεργασία που την δημιουργήσε.
- Συνήθως, μια γονική διαδικασία δημιουργεί μια διοχέτευση και την χρησιμοποιεί για να επικοινωνεί με μια διεργασία που δημιούργησε.
- Επώνυμες διοχετεύσεις (Named pipes) – είναι προσβάσιμες χωρίς να υπάρχει σχέση γονέα-παιδιού ή παιδιών ίδιου πατέρα.
- Οι κοινές διοχετεύσεις επιτρέπουν την επικοινωνία σε τυπικό τρόπο παραγωγού-καταναλωτή
- Ο παραγωγός γράφει στο ένα άκρο (άκρο εγγραφής write-end)
- Ο καταναλωτής διαβάζει από το άλλο άκρο (άκρο ανάγνωσης read-end)
- Επομένως, οι κοινές διοχετεύσεις είναι μονής κατεύθυνσης
- Απαιτείται σχέση γονέα-παιδιού μεταξύ των διεργασιών επικοινωνίας
- Στα Windows καλούνται ως ανώνυμες διοχετεύσεις anonymous pipes

Τρόπος εγγραφής/ανάγνωσης: **μονόδρομος**! Πάντα γράφω από το ένα άκρο και διαβάζω από το άλλο. Όποιος ξέρει το άκρο εγγραφής της διοχέτευσης μπορεί να γράφει και όποιος ξέρει το άκρο ανάγνωσης μπορεί να διαβάζει.

- 1) Όταν διαβάζω κάτι, αφαιρείται από την σωλήνωση.
- 2) Όταν γράφω, τα δεδομένα που γράφω μπαίνουν στο τέλος και όταν θέλω να διαβάσω από την σωλήνωση θα διαβάσω αυτό που έχει μπει πρώτο (ουρά).
- 3) Όταν πάω να διαβάσω κάτι, ενώ δεν υπάρχει τίποτα για να διαβάσω, το πρόγραμμα κολλάει. Έτσι, γίνεται ο συγχρονισμός (π.χ. μεταξύ των παιδιών και του πατέρα).

Η κλήση συστήματος **pipe()** δημιουργεί μια διοχέτευση, ένα κανάλι δεδομένων μονής κατεύθυνσης που μπορεί να χρησιμοποιηθεί για επικοινωνία μεταξύ διεργασιών. Ο **πίνακας fd** χρησιμοποιείται για την επιστροφή δύο περιγραφητών αρχείων που αναφέρονται στα άκρα της διοχέτευσης. Το στοιχείο **fd[0]** του πίνακα αναφέρεται στο άκρο ανάγνωσης της διοχέτευσης και το **fd [1]** αναφέρεται στο άκρο εγγραφής.

Στο pipe βάζουμε τη διεύθυνση ακεραίου.

Παράδειγμα εκτέλεσης προγράμματος

```
./ask3 n - -round-robin ή ./ask3 n - -random
```

Το όρισμα **n** πρέπει να είναι ακέραιος θετικός αριθμός που προσδιορίζει το πλήθος των διεργασιών παιδιών που θα δημιουργήσει ο πατέρας.

Το όρισμα **round-robin** ή **random** προσδιορίζει τον τρόπο με τον οποίο η διεργασία πατέρας μοιράζει τις εργασίες:

round-robin: με κυκλική επαναφορά **random**: με τυχαίο τρόπο

Η διεργασία πατέρας μπορεί να δεχθεί από τον χρήστη μέσω του terminal τις παρακάτω εντολές

- **help**. Στην περίπτωση αυτή η διεργασία πατέρας εκτυπώνει το μήνυμα: Type a number to send job to a child!
- **exit**. Στην περίπτωση αυτή η διεργασία πατέρας τερματίζει αφού πρώτα τερματίσει όλα τα παιδιά εκτυπώνοντας κατάλληλα μηνύματα π.χ. All children terminated
- Έναν ακέραιο αριθμό π.χ. 12. Στην περίπτωση αυτή η διεργασία πατέρας διαλέγει μια διεργασία παιδί και της αναθέτει την εργασία στέλνοντας της τον αριθμό μέσω μιας διοχέτευσης και εκτυπώνει το μήνυμα [Parent] Assigned 12 to child 5
- Σε κάθε άλλη περίπτωση να εκτυπώνεται το μήνυμα Type a number to send job to a child!

Κάθε **διεργασία παιδί** μπορεί να δεχθεί από την διεργασία πατέρα μέσω μιας διοχέτευσης ένα αριθμό τον οποίο αυξάνει κατά ένα, περιμένει για 5 secs και τον επιστρέφει στην διεργασία πατέρα μέσω μιας άλλης διοχέτευσης. Αυτό γίνεται επαναληπτικά μέχρι να την τερματίσει η διεργασία πατέρας.

Συνεπώς η διεργασία πατέρας μπορεί να δεχθεί δεδομένα από ένα σύνολο διοχετεύσεων ή από το terminal, αυτό είναι εφικτό να γίνει μέσω της **select()**.

Η `select()` επιτρέπει σε ένα πρόγραμμα να παρακολουθεί πολλούς περιγραφητές αρχείων, έως ότου σε ένα ή περισσότερους από τους περιγραφητές αρχείων γίνει δυνατή η εκτέλεση μιας λειτουργίας εισόδου / εξόδου (π.χ. ανάγνωση ή εγγραφή) χωρίς αποκλεισμό.

Στον πατέρα έχουμε τη `select()` γιατί πρέπει να διαχειριστεί η διοχετεύσεις και του πληκτρολογίου, ενώ το παιδί έχει να διαχειριστεί μόνο ένα πράγμα.

- Για τυχαία παραγωγή αριθμών που είναι πιθανό να σας χρειαστεί στον διαμοιρασμό των εργασιών με random τρόπο προτείνεται η `rand()`. Η συνάρτηση `rand()` επιστρέφει έναν ψευδο-τυχαίο ακέραιο στο εύρος 0 έως `RAND_MAX`
- Για δυναμική διάθεση μνήμης που είναι πιθανό να σας χρειαστεί για τον ορισμό πινάκων αφού διαβάσετε το η από την γραμμή εντολών προτείνεται η `malloc()`. Η συνάρτηση `malloc()` κατανέμει bytes μεγέθους και επιστρέφει έναν δείκτη στην κατανεμημένη μνήμη. Η μνήμη δεν αρχικοποιείται. Εάν το μέγεθος είναι 0, τότε η `malloc()` επιστρέφει είτε `NULL`, είτε έναν μοναδικό δείκτη τιμή δείκτη που μπορεί αργότερα να μεταβιβαστεί με επιτυχία στην `free()`.

`read(parent_to_child_fd[i][0], &value, sizeof(int))`

Η `read()` προσπαθεί να διαβάσει έως και `int` bytes από τον περιγραφέα αρχείου `fd` στο buffer που αρχίζει από το `value`.

`write(child_to_parent_fd[i][1], &value, sizeof(int))`

Η `write()` γράφει μέχρι και `int` bytes από το buffer ξεκινώντας από το `value` στο αρχείο στο οποίο αναφέρεται ο περιγραφέας αρχείου `fd`.