

1. Αρχεία.

- Έχουμε 2 αρχεία κώδικα το Pm.c και το Pc.c.
- Το Pm.c είναι το κύριο πρόγραμμα το οποίο δημιουργεί το ergasia2
- Το Pc.c είναι παιδί του Pm.c το οποίο κάνει τους υπολογισμούς.
- Και έχουμε και το ergasia2.h το οποίο περιέχει τις δηλώσεις και για τα 2 προγράμματα.
(Στην περίπτωση αυτή έχω μόνο το τι περιέχουν τα μηνύματα που στέλνονται μέσω pipes)
- Makefile με τους αυτοματισμούς.

2. Δομή.

- Ξεκινάει η Pm τσεκάροντας την είσοδο των arguments, δημιουργεί τους πίνακες για τα μεγέθη, τους μετρητές (K) εντός του κύκλου και (T) συνολικών σημείων.
- Επίσης εδώ δημιουργούμε(δηλώνουμε) 2 πίνακες για τα pipes (fdin,fdout).
- Στην συνέχεια δημιουργούμε αντίγραφα των διεργασιών και κατασκευάζουμε 2 pipes για κάθε διεργασία παιδί ξεχωριστά, έναν για την αποστολή μηνυμάτων και έναν για την λήψη.
- Κάθε διεργασία παιδί στέλνει μέσω ροής (stdin-stdout) την πληροφορία του στην σωλήνα εισόδου και εξόδου αντίστοιχα. Ετσι γράφοντας πχ στο stdout, γράφει στο σωλήνα(pipe).
- Στην συνέχεια μπαίνουμε σε μια λούπα for, όπου τρέχουμε τα τέστ, παράγουμε τυχαία σημεία και τα στέλνουμε στα παιδιά μέσω pipes. Μετά διαβάζουμε τις απαντήσεις.
- Η Μαμά διεργασία μετα την λήψη των αποτελεσμάτων, ξεμπλοκάρει και περνάει τα αποτελέσματα στους πίνακες K και T.
- Όταν έχουν ολοκληρωθεί όλα τα τέστ, η Μαμά στέλνει μήνυμα μεγέθους 0 στο παιδί ότι ολοκληρώθηκε η διαδικασία. Υπολογίζει όλα τα αποτελέσματα (η Pm) και ολοκληρώνει.
- Τέλος η Pc Διεργασία ξεκινάει και μπαίνει σε έναν ατέρμων βρόγχο, όπου προσπαθεί να διαβάσει το μήνυμα (0) από την ροή (stdin), μόλις το σήμα αφιχθεί υπολογίζει τα αποτελέσματα και τα γράφει στην έξοδο της ροής(stdout).
- Εκτυπώσεις.
- Αποδεσμεύσεις.

```
konstantinoskonstantinakos@ubuntu: ~/Desktop/pipes_pi
konstantinoskonstantinakos@ubuntu:~/Desktop/pipes_pi$ ./ergasia2pipes 10 99 100000
Tests: 100000
T(10)=49929 T(99)=50071
K(10)=39153 K(99)=39270
P(10) = 3.14
P(99) = 3.14
konstantinoskonstantinakos@ubuntu:~/Desktop/pipes_pi$ ./ergasia2pipes 88 99 100 99999
Tests: 99999
T(88)=33398 T(99)=33096 T(100)=33505
K(88)=26398 K(99)=25971 K(100)=26338
P(88) = 3.16
P(99) = 3.14
P(100) = 3.14
konstantinoskonstantinakos@ubuntu:~/Desktop/pipes_pi$ ./ergasia2pipes 7 8 9 1000000
Tests: 1000000
T(7)=332758 T(8)=333981 T(9)=333261
K(7)=261303 K(8)=262147 K(9)=261557
P(7) = 3.14
P(8) = 3.14
P(9) = 3.14
konstantinoskonstantinakos@ubuntu:~/Desktop/pipes_pi$ ./ergasia2pipes
LATHOS XRHSH: ./ergasia2pipes <size> [ARGS?.] <num_tests?>
konstantinoskonstantinakos@ubuntu:~/Desktop/pipes_pi$
konstantinoskonstantinakos@ubuntu:~/Desktop/pipes_pi$ make clean
rm -rf ergasia2pipes Pc
konstantinoskonstantinakos@ubuntu:~/Desktop/pipes_pi$
```

Πηγές.

Stackoverflow

<https://stackoverflow.com/questions/33884291/pipes-dup2-and-exec>

Pipe() <https://www.geeksforgeeks.org/pipe-system-call/>

<https://man7.org/linux/man-pages/man2/pipe.2.html>

fileno() <https://man7.org/linux/man-pages/man3/fileno.3.html>

dup() dup2() <https://www.geeksforgeeks.org/dup-dup2-linux-system-call/>

<https://linux.die.net/man/2/dup2>

<https://man7.org/linux/man-pages/man2/dup.2.html>

FILEDESCRIPTOR YOUTUBE

<https://www.youtube.com/watch?v=5fnVr-zH-SE>

Two way communication between processes (using pipes) in C

<https://www.youtube.com/watch?v=8Q9CPWuRC6o><https://stackoverflow.com/questions/33884291/pipes-dup2-and-exec>

<http://unixwiz.net/techtips/remap-pipe-fds.html>