

# Λειτουργικά Συστήματα

## Εργασία 1η

Κουλούρης Διονύσιος, 3160069  
Παπαδόπουλος Γεώργιος, 3160131

### Αρχείο headers/common:

Στο αρχείο common έχουμε υλοποιήσει τις συναρτήσεις readInput, split\_Args, getFilenames, exe και SplitPipes.

Η readInput διαβάζει και επιστρέφει την είσοδο που θα δώσει ο χρήστης από το πληκτρολόγιο.

Η split\_args χωρίζει το κείμενο που έχει επιστρέψει η readInput σε έναν NULL terminated πίνακα, για να μπορέσει να χρησιμοποιηθεί σωστά στην εκτέλεση αργότερα.

Η getFilenames δέχεται ως όρισμα την εξοδο του split args και έναν χαρακτήρα (η τον ">" η τον "<"). Ψαχνει να βρει τον χαρακτήρα αυτό στον πίνακα με τα arguments, και εαν τον βρεί, επιστρέφει το επόμενο argument, που θα είναι το όνομα του αρχείου.

Η exe δεχεται ως όρισμα τον πίνακα των arguments, κανει fork την current διεργασία, εαν το fork πετυχει εκτελεί την διεργασία με την execvp και περιμένει η διεργασία αυτη να τελειώσει.

Η splitPipes χωρίζει τις εντολές με βάση τα pipes ("|") Που τις χωρίζουν.

### Ερώτημα 1:

Σε αυτο το ερώτημα, διαβάζουμε τα δεδομένα απο τον χρήστη, εαν δώσει ενα argument το εκτελουμε, αλλιως πεταει error οτι εχει δώσει περισσότερα απο ενα και εκτελεί μόνο το 1ο.

### Ερώτημα 2:

Σε αυτο το ερώτημα, διαβάζουμε τα δεδομένα απο τον χρήστη. Χρησιμοποιούμε την dup και την dup2 με εισόδους 0 και 1. Διαβάζουμε την είσοδο και ελεγχουμε εαν υπάρχει < η > στις εντολες μας. Εαν υπάρχουν,

ανοιγουμε τα files με το `fopen` ή/και τα κλείνουμε με το `close(out)`, αφού πρώτα το έχουμε ανοίξει για `write_only`. Στο τέλος κάνουμε τον αντίστοιχο έλεγχο για την μία εντολή.

### **Ερώτημα 3:**

Σε αυτο το ερώτημα κάνουμε ότι και στο 2, με τη διαφορά ότι δεν υπάρχει ο έλεγχος για την μία εντολή.

### **Ερώτημα 4:**

Σε αυτο το ερώτημα κάνουμε ότι και στο 3 και επιπλέον: Χωρίζουμε τις εντολές στα 2 (δεξια και αριστερά του `pipe`). Ο πίνακας `pipefd` χρησιμοποιείται για να μπουν τα `input` και `output` των 2 εντολών. Χωρίζουμε τα `arguments` Καθε εντολής και την εκτελούμε ξεχωριστά σε μια `for` και στο τέλος κανουμε `close` τα `pipefd`. Υπαρχει και ελεγχος για την περιπτωση να εχουμε δωσει εντολη χωρις `pipe`, οποτε γλυτωνουμε την ολη αυτη διαδικασία.

### **Ερώτημα 5:**

Σε αυτο το ερώτημα εχουμε υλοποιησει μια πιο γενικευμενη εκδοση του ερωτήματος 4. Υπαρχει πίνακας που αποθηκευονται τα `fds`, 2 για καθε εντολη. Στη συνεχεια εκτελείται η `dup2` και ανάλογα με το αν είναι πρώτη η όχι δέχεται το κατάλληλο `pipe fd` ως είσοδο. Στη συνέχεια γίνονται όλα τα `pipe fds close`. Υπαρχει ανάλογος έλεγχος για το εαν υπάρχουν `pipes`.