

Προγραμματιστική Άσκηση 2

Προγραμματισμός Συστήματος

Όνομα: Βασιλική Γεωργάρα

A.M.: 1115201400026

Η εργασία μου έχει υλοποιηθεί σε γλώσσα C.

Από την πρώτη άσκησή μου έχω χρησιμοποιήσει τη λίστα λέξεων και το Trie.

Η διαφορά της posting list σε σχέση με την 1^η εργασία είναι ότι μας λέει σε ποιο αρχείο βρίσκεται η λέξη, σε ποια γραμμή του αρχείου αυτού και με τι συχνότητα σε αυτή τη γραμμή σε αυτό το αρχείο. Από εκεί και πέρα όλα τα υπόλοιπα σημεία είναι ίδια σε σχέση με την 1^η. Επίσης, άφησα συναρτήσεις όπως η «marked», οι οποίες δεν χρησιμοποιούνται, απλά θεώρησα πιο δύσκολο να βγάλω κώδικα, από το να τον αφήσω και να μην τον χρησιμοποιήσω. Στη λίστα ListChar μπαίνουν τα paths και ύστερα οι λέξεις που βρίσκονται στα txt.

Στο Trie έχω κρατήσει τη df στην οποία έχω κάνει προσαρμογές. Κάνει την αναζήτηση της λέξης και για κάθε κόμβο της posting list μου επιστρέφει το αντίστοιχο αποτέλεσμα.

Ο JobExecutor για αρχή διαβάζει τα ορίσματα, εγκαθιστά τον handler για το sigchld (ο οποίος κάνει respond το παιδί με το που θα πεθάνει) και στη συνέχεια διαβάζει τα directories του αρχείου dirs και τα βάζει στη λίστα. Η λίστα αυτή είναι global γιατί όταν πεθάνει ένα παιδί, ο handler δεν μπορεί να πάρει ορίσματα οπότε και δεν θα έβλεπε τη λίστα αν δεν ήταν global ώστε να τη μεταβιβάσει στο καινούργιο παιδί που θα γεννηθεί.

Στη συνέχεια κάνω malloc τους global πίνακες, σβήνει και φτιάχνει όσα pipes χρειάζεται ο JobExecutor, τα ανοίγει και μετά μπορεί να φτιάξει τους Workers γιατί ο καθένας πλέον μπορεί να ανοίξει το pipe του. Στέλνω κυκλικά τα folders και μόλις σταλούν όλα στέλνω το «end» σε όλους τους workers ώστε να ξέρουν ότι η διανομή τελείωσε. Οπότε εκτυπώνω το prompt ώστε να διαβάσω εντολές και τις διαβάζω.

Αν είναι exit, ακυρώνω το handler για να μην κάνει την αναπλήρωση μόλις το παιδί τελειώσει, δηλαδή κάνει ignore το σήμα sigchld. Το στέλνει και παίρνει το reply δηλαδή πόσα keywords έχουνα βρεθεί.

Αν είναι wc, στέλνω την εντολή και προσθέτω στις ολικές μεταβλητές και τυπώνω το τελικό αποτέλεσμα.

Αν είναι search, επαναφέρω το κενό, στέλνω την εντολή σε όλους και για έναν έναν παίρνω τις απαντήσεις. Οι απαντήσεις είναι είτε «end», είτε «endD», η πρώτη αν είναι κανονικό, η

2^η αν είναι από το deadline, δηλαδή forced exit. Στο τέλος εκτυπώνω πόσοι απάντησαν on time.

Αν είναι mincount, τότε επαναφέρω το κενό, γράφω και λαμβάνω δύο πράγματα. Τον current αριθμό ελάχιστο και το filename όπου θέλω το συνολικό ελάχιστο οπότε θέτω στο αρχικό έναν τεράστιο αριθμό και ελέγχω κάθε φορά αν ο current είναι καλύτερος από το minimum. Αν είναι κρατάω το minimum. Ομοίως στο maxcount, αλλά το ανάποδο.

Μόλις βγω από το loop, κάνω wait τα παιδιά, κλείνω τα pipe και τα σβήνω.

Η συνάρτηση sighandlerCHLD καλείται όταν ένα παιδί έχει τελειώσει. Ψάχνει το παιδί που σκοτώθηκε και μόλις βρει τη θέση του, αδειάζει τα buffer και φτιάχνω μια νέα διεργασία στη θέση που σκοτώθηκε.. Στέλνω τα δικά του directories και τελειώσε.

Ο worker έχει τον δικό του handler ώστε να διαχειρίζεται το CanWrite και την απάντηση. Με το που ξεκινάει φτιάχνει το handler για το sigalarm. Φτιάχνει τα pipes και λαμβάνει μηνύματα μέχρι να λάβει «end». Φτιάχνει το log και το κλείνει γιατί αν τυχόν πεθάνει ένα παιδί και δεν έχω κάνει close, ό,τι έχω γράψει ως τότε θα χαθεί. Μόλις τελειώσει αυτό, περιμένει μία μία τις εντολές, βλέπει την 1^η λέξη της εντολής και:

Αν είναι exit, στέλνει πίσω τα πόσα strings βρήκε.

Αν είναι wc, στέλνει το μήνυμα που πρέπει.

Αν είναι search, πρώτα κοιτάει αν έχει deadline. Αν υπάρχει κάτι μετά το -d, το πρόγραμμα θα σκάσει. Παίρνω την τιμή του deadline και το κάνω alarm,εκτελώ τον handler ο οποίος θα κάνει το CanWrite 0 και θα στείλει «endD». Άρα κανονικά, αν δεν φτάσω τα 10 ορίσματα, βάζω στη df την εκάστοτε λέξη και μου επιστρέφει 1 αν τη βρήκε ή 0 αλλιώς. Γράφω το searchString γιατί αργότερα θέλω να το εκτυπώσω και πάω στην επόμενη λέξη. Σε περίπτωση που όσο κάνω αυτή τη δουλειά έρθει το sigalarm, το CanWrite γίνεται 0 (όπου υπάρχουν χρονοβόρα κομμάτια κώδικα υπάρχουν ελέγχοι για το CanWrite), φεύγει από την επανάληψη και πάει στην επόμενη εντολή. Αν βγει κανονικά από την επανάληψη τότε γράφει το «end», μηδενίζει το alarm έτσι ώστε να μην περιμένει σε περίπτωση που τελειώσει πιο νωρίς, και κάνει unblock τα σήματα. Στο τέλος γράφω στο log το Timestamp(έχει παρθεί από το ίντερνετ).

Αν είναι mincount, κάνει search και θα επιστρέψει posting list, που θα έχει filename, γραμμή και συχνότητα. Για κάθε κόμβο, κοιτάω πού αλλάζει το filename γιατί εκεί που αλλάζει εξετάζω αν το σύνολο των συχνοτήτων είναι καλύτερο από το ελάχιστο που έχουμε εμείς τότε δεν το κρατάω. Αν δεν έχει posting list το minimum μένει 0, γράφω στο log και στέλνω την απάντηση πίσω. Το ίδιο και το maxcount.

Τα parsing των αρχείων γίνονται με τις συναρτήσεις που καλούνται στην αρχή.

