

ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΣΥΣΤΗΜΑΤΟΣ – ΕΡΓΑΣΙΑ 2

ΠΛΥΜΜΕΝΟΣ ΚΩΝΣΤΑΝΤΙΝΟΣ – 1115201500133

Το Project αυτο περιεχει τους εξης φακελους :

src/ : περιεχει τα αρχεια πηγαιου κωδικα

include/ : περιεχει τα αρχεια επικεφαλιδας

bin/ : σε αυτον τον φακελο δημιουργουνται τα object files

build/ : σε αυτον τον φακελο δημιουργειται το εκτελεσιμο

scripts/ : περιεχει τα scripts που ζητουνται

log/ : σε αυτον τον φακελο δημιουργουνται τα logfiles της εφαρμογης

documentation/ : περιεχεται το documentation του project

datasets/ : περιεχονται αρχεια και φακελοι με τα οποια εγινε το testing της εφαρμογης

Παρεχεται Makefile στο οποιο γινεται seperate compilation

Η εκτελεση : ./build/JobExecutor -d docfile -w N οπου N ο αριθμος των workers.

Επικοινωνια διεργασιων :

Η επικοινωνια μεταξυ του JobExecutor και των παιδιων του γινεται μεσω 2*αριθμος παιδιων named pipes τα οποια ειναι blocking. Στις πρωτες μισες ο JobExecutor γραφει και τα παιδια διαβαζουν και στις υπολοιπες το αντιθετο. Blocking pipes σε συνδυασμο με τον καταλληλο χειρισμο της συναρτησης poll() επιτυγχανουν την αναμενομενη επικοινωνια.

Επισης για τον χειρισμο των εντολων του προγραμματος εχει οριστεί ενα πρωτοκολλο επικοινωνιας μεταξυ του πατερα και των παιδιων. Συγκεκριμενα εχουν οριστεί σταθερου μηκους Headers τα οποια υποδεικνυουν την εντολη που πρεπει να εκτελεσται. Αν δεν υπαρχει κατι να επιστραφει σαν αποτελεσμα (πχ στην μεταβιβαση των paths) στελνεται ενας κωδικος DONE.

Χειρισμός Σημάτων :

Η εφαρμογή διαχειρίζεται τα εξής signals :
SIGCHLD , SIGINT , SIGQUIT , SIGTERM

Στην περίπτωση που ληφθεί σήμα SIGCHLD ο handler του σήματος κάνει σέτ μια global μεταβλητή η οποία ελέγχεται από το main loop του JobExecutor για να εξεταστεί αν κάποιο παιδί έχει πεθάνει. Αν η μεταβλητή είναι set τότε ο JobExecutor με την βοήθεια ενός πίνακα με τα pid των παιδιών του και με βάση το index τους στον πίνακα αντικαθιστά τον προβληματικό worker με έναν καινούριο στον οποίο μεταβιβάζει τα paths του παλιού.

Στην περίπτωση που ληφθεί σήμα SIGINT | SIGTERM | SIGQUIT ο handler των σημάτων αυτών αγνοεί την default λειτουργία τους, ενημερώνει καταλλήλα τον χρήστη και γίνεται ομαλός τερματισμός της εφαρμογής σαν να είχε δώσει ο χρήστης την εντολή exit.

Αποθήκευση και Χειρισμός Δεδομένων :

Η αποθήκευση των δεδομένων γίνεται με την χρήση ενός trie (από την προηγούμενη εργασία αλλά ανανεωμένο) και με την χρήση δυναμικών πινάκων. Συγκεκριμένα αποθηκεύονται όλα τα paths που διαχειρίζεται ο worker των οποίων το id τους είναι το index που έχουν στον πίνακα. Το ίδιο γίνεται με τα ονόματα των αρχείων. Υπάρχει ένας πίνακας που κάνει mapping τα paths στα αρχεία και ένας πίνακας που αποθηκεύει το περιεχόμενο των αρχείων. Έτσι σε κάθε node του postinglist μιας λέξης αποθηκεύεται τα identifiers του αρχείου και του μονοπατιού.

Επίσης σε κάθε node του postinglist αποθηκεύεται μια ακόμα λίστα που περιέχει τους αριθμούς των γραμμών που βρίσκεται η λέξη στο αρχείο και το πρώτο byte της γραμμής ώστε να επιστρέφεται άμεσα η γραμμή που περιέχει την λέξη.

Bash Scripts :

Υλοποιούνται 3 scripts ενα για καθε ερωτημα. Τα 3 scripts διαβαζουν τα αρχεια απο τον καταλογο log. Για την ευρεση του αριθμου των αναζητησεων απλα παιρνει το column με την εντολη και εξεταζει αν αυτη ηταν /search και αυξανει εναν μετρητη. Στα αλλα δυο scripts απομονωνονται τα πεδια που περιεχουν την λεξη που αναζητηθηκε και τα μονοπατια που βρεθηκαν με την χρηση της cut και τα αποτελεσματα συνενωνονται με την εντολη paste. Στην συνεχεια με την χρηση της awk για καθε λεξη μετριουνται ποσα μονοπατια εχουν βρεθει και κραταμε μονο τις λεξεις που εχουν βρεθει εστω κ μια φορα. Τελος γινεται σορτ με παραμετρο -u για να εξαλειψουμε τα διπλοτυπα και αναλογα αν θελουμε το μεγαιστο η το ελαχιστο χρησημοποιουμε την head/tail.