

Ονοματεπώνυμο: Θεοφανόπουλος Μιχαήλ
Αριθμός Μητρώου: 1115201800053
2η Εργασία Λειτουργικά Συστήματα

Η εργασία χωρίζεται στα εξής εκτελέσιμα αρχεία:

- `functions.cpp` , το οποίο περιέχει τις συναρτήσεις που χρησιμεύουν για την εύρεση των πρώτων αριθμών καθώς την `findPrimes` η οποία χρησιμοποιεί κάποια από τις συναρτήσεις αυτές για την εύρεση των αριθμών και τέλος την `signalFunction` που έχει δημιουργηθεί για το πιάσιμο `signals` τύπου `SIGUSR1`
- `list.cpp` , το οποίο αναπάριστα μια λίστα με συγκεκριμένες λειτουργίες και συγκεκριμένη υλοποίηση, όπως δηλαδή έκρινα ότι θα την χρειαστώ μέσα στο πρόγραμμα μου.
- `myprimes.cpp` , το οποίο αναπαριστά τον C κόμβο του σχήματος που υπάρχει στην εκφώνηση. Σκοπός της συγκεκριμένης `main` είναι να δημιουργεί `numOfChildren L nodes` και να διαβάζει τα δεδομένα από αυτούς. Επίσης πιάνει `signals` τύπου `SIGUSR1` από τους `workers` της.
- `l.cpp` , το οποίο αναπαριστά τον εκάστοτε `L` κόμβο του σχήματος. Σκοπός του κάθε κόμβου είναι να δημιουργεί τους `workers`, να διαβάζει τα αποτελέσματα από αυτούς και να τα στέλνει στην ρίζα.
- `w.cpp` , το οποίο αναπαριστά τους `workers`, οι οποίοι χρησιμοποιώντας την σωστή `primes function` βρίσκουν σωστά τους πρώτους αριθμούς στο δοθέντα διάστημα και τους στέλνουν στους `parent nodes`.

Η επικοινωνία μεταξύ των διεργασιών επιτυγχάνεται μέσω `pipes`. Κάθε `child process` επικοινωνεί με το `parent process` μέσω ξεχωριστού `pipes` από όλα τα υπόλοιπα. Οι `workers` στέλνουν τα αποτελέσματα που βρίσκουν ξεχωριστά ένα ένα στους `L nodes`, οι οποίοι τα αποθηκεύουν προσωρινά σε μία δυναμική λίστα από όπου τα περνάνε σε έναν στατικό πίνακα και τα στέλνουν στην ρίζα, η οποία κάνει τον τελικό απολογισμό των αποτελεσμάτων.

Για την δημιουργία των `child processes`, σε κάθε `level` του δέντρου διεργασιών (εκτός φυσικά από τα φύλλα-`W`), χρησιμοποιείται ένας πίνακας χαρακτήρων `vector` ο οποίος περιέχει τα `arguments` που θα χρειαστεί η `main` του `child process` για να εκτελεστεί (σύμφωνα πάντα με τις σχεδιαστικές επιλογές που έχω ακολουθήσει στην υλοποίηση μου και περιγράφονται καλύτερα σε σχόλια μαζί με τον κώδικα). Στη συνέχεια, εκτελείται η εντολή `execvp` με `input` τον `vector` και έτσι λειτουργεί η διασύνδεση των διεργασιών. Η παραπάνω διαδικασία γίνεται για κάθε ένα από τα `numOfChildren child processes` που πρέπει να δημιουργηθούν. Επέλεξα την παραπάνω υλοποίηση καθώς θεώρησα πως απευθύνεται καλύτερα στο ζητούμενο της άσκησης.

Στο τέλος του προγράμματος αποδεσμεύεται σωστά κάθε μνήμη που έχει δεσμευτεί δυναμικά.

Υπάρχει το `makefile`, με την χρήση του οποίου μπορεί να μεταγλωττιστεί το πρόγραμμα με την εντολή `$make all` και για να τρέξει πρέπει να δοθεί η εντολή `$/myprimes -l (input) -u (input) -w (input)`.

Επέλεξα να υλοποιήσω την άσκηση σε `C++` καθώς είμαι περισσότερο εξοικειωμένος με την χρήση της, και καθώς δίνεται η δυνατότητα να χρησιμοποιηθούν κλάσεις και διάφορες χρήσιμες βιβλιοθήκες της `C++`.