

## Αναφορά 1ης Εργαστηριακής Άσκησης

Το 1ο μέρος της εργαστηριακής άσκησης, ζητούσε την υλοποίηση ενός line based text editor, το οποίο θα υλοποιείται ως μια δυναμικά υλοποιημένη, διπλά συνδεδεμένη λίστα γραμμών με διάφορες εντολές για την επεξεργασία του κειμένου.

Ανάλυση Υλοποίησης του κειμενογράφου και των επιμέρους εντολών του:

Αρχικά δημιουργήθηκε το στοιχείο που περιέχεται στη λίστα(node) που έχει μεταβλητές την σειρά της λίστας και δείκτες στο επόμενο και προηγούμενο στοιχείο. Στη συνέχεια, δημιουργήθηκε μια κλασσική διπλά συνδεδεμένη λίστα(με στοιχείο το node), με τις κλασσικές μεθόδους: (insert at back, insert after, insert, delete) αλλά και η print, για εκτύπωση ενός στοιχείου τη λίστας, print list, για εκτύπωση όλης της λίστας και print total lines and chars. Τέλος χρησιμοποιούμε την κλάση my Editor για τη δημιουργία του menu και την επικοινωνία της λίστας με τον χρήστη.

Περιγραφή υλοποίησης των εντολών του κειμενογράφου:

- ^: Τοποθετεί στο δείκτη που διαβάζει το αρχείο το δείκτη του πρώτου στοιχείου της λίστας και ανανεώνει την τιμή του current line.
- \$: Τοποθετεί στο δείκτη που διαβάζει το αρχείο το δείκτη του τελευταίου στοιχείου της λίστας και ανανεώνει την τιμή του current line.
- -: Αν δεν είναι το πρώτο στοιχείο της λίστας ή αν δεν είναι κενός, ο δείκτης που διαβάζει τη λίστα χρησιμοποιώντας την getPrev(), η τιμή του αλλάζει στο προηγούμενο στοιχείο της λίστας.
- +: Αν δεν είναι το τελευταίο στοιχείο της λίστας ή αν δεν είναι κενός, ο δείκτης που διαβάζει τη λίστα χρησιμοποιώντας την getNext(), η τιμή του αλλάζει στο προηγούμενο στοιχείο της λίστας.
- Οι εντολές t, a εισάγουν μια νέα γραμμή (που δίνει ο χρήστης) στη λίστα πριν ή μετά από τη θέση που βρίσκεται ο δείκτης αναζήτησης της λίστας. Υλοποιούνται με τις εντολές insert at back και insert after.
- d: Διαγράφει τον κόμβο στον οποίο βρίσκεται ο δείκτης αναζήτησης της λίστας, υλοποιείται με την delete, η οποία επιστρέφει την νέα θέση του δείκτη αναζήτησης του αρχείου μέσα στη λίστα.
- l: Εκτυπώνει όλα τα στοιχεία που βρίσκονται στη λίστα με την εντολή printlist(), η οποία καθώς διασχίζει όλη τη λίστα εκτυπώνει τη σειρά του κάθε στοιχείου της λίστας.
- n: Μας επιτρέπει να ενεργοποιήσουμε ή να απενεργοποιήσουμε την εμφάνιση του αριθμού της γραμμής, όταν αυτή εκτυπώνεται.
- p: Εκτυπώνει τη γραμμή του κόμβου που βρίσκεται ο δείκτης αναζήτησης και υλοποιείται με την εντολή print() .
- q: Δίνει στην boolean μεταβλητή exit την τιμή true, με αποτέλεσμα να τερματίζει το πρόγραμμα.
- x: Ίδια λειτουργικότητα με την q, αλλά πριν τερματίσει το πρόγραμμα σώζει τη λίστα στο αρχείο.
- =: Εκτυπώνει τον αριθμό της γραμμής που βρισκόμαστε(print int current line).
- #: Διασχίζει όλη τη λίστα, διαβάζει κάθε γραμμή, υπολογίζει το μήκος της και στη συνέχεια, προσθέτει και εκτυπώνει τα μήκη όλων των γραμμών και των συνολικών γραμμών του κειμενογράφου.

Το δεύτερο μέρος της άσκησης ζητούσε τη δημιουργία ενός δυαδικού αρχείου δεικτοδότησης από τις λέξεις του κειμενογράφου και τη δυνατότητα εύρεσης λέξεων μέσα σε αυτό το αρχείο και την εκτύπωση των γραμμών που βρίσκονται οι λέξεις αυτές.

Ανάλυση Υλοποίησης του δυαδικού αρχείου και των επιμέρους εντολών του:

Αρχικά δημιουργήθηκε η κλάση tuple, που αποτελείται από ένα token και έναν αριθμό και χρησιμοποιείται για την αποθήκευση των λέξεων και των γραμμών που βρίσκονται στο δυαδικό αρχείο. Στη συνέχεια δημιουργήθηκε η κλάση MyTupleEditor, η οποία υλοποιεί τις μεθόδους createTupleArray(), writeToDisk(), readFromDisk(), readSerial(), readbinary(). Τέλος προστέθηκαν στο MyEditor ένα instantiate του MyTupleEditor, καθώς και οι λειτουργικότητες του.

Περιγραφή υλοποίησης των εντολών του δυαδικού αρχείου δεικτοδότησης:

- c: Δημιουργεί ένα ταξινομημένο TupleArray από τη διπλά συνδεδεμένη λίστα(createTuplesArray), με στοιχεία που πληρούν τα κριτήρια, και στη συνέχεια ο πίνακας αυτός γράφεται στο δίσκο ανά σελίδες(writeToDisk()), αφού πρώτα τα στοιχεία έχουν μετατραπεί σε bytes. Τέλος εκτυπώνεται πόσες σελίδες δημιουργήθηκαν.
- v: Εκτυπώνει το περιεχόμενο του αρχείου δεικτοδότησης(readFromDisk()), διαβάζοντας σελίδα σελίδα το αρχείο δεικτοδότησης και μετατρέποντας τα δεδομένα από μορφή bytes σε χαρακτήρες και ακεραίους.
- s: Ο χρήστης γράφει τη λέξη που ψάχνει και μετά αυτή η λέξη αναζητάται σειριακά μέσα στο δυαδικό αρχείο. Αν βρεθεί εκτυπώνεται η λέξη και το πόσες φορές βρέθηκε στο αρχείο. Αν δεν υπάρχει η λέξη στο αρχείο, η αναζήτηση σταματά, όταν βρεθεί μια λεξικογραφικά μεγαλύτερη λέξη από αυτή που αναζητείται. Σε κάθε περίπτωση εκτυπώνονται οι προσβάσεις στο αρχείο.
- b: Ο χρήστης γράφει τη λέξη που ψάχνει και μετά αυτή η λέξη αναζητάται μέσα στο δυαδικό αρχείο χρησιμοποιώντας τον αλγόριθμο δυαδικής αναζήτησης. Στην περίπτωση που βρεθεί η λέξη, ελέγχουμε πριν και μετά τη λέξη, αν αυτή βρίσκεται και σε άλλες γραμμές.

Πείραμα

Μέσος όρος προσβάσεων για τις γνωστές λέξεις:

testfile\_x2.txt

serial search:118.25, binary search:17.50

testfile\_x5.txt

serial search:294.25, binary search:28

testfile\_x10.txt

serial search:587.50, binary search:28.25

testfile\_x1000.txt

serial search:58,650, binary search:1076.50

Μέσος όρος προσβάσεων για τις άγνωστες λέξεις:

testfile\_x2.txt

serial search:140.90, binary search: 16.93

testfile\_x5.txt

serial search:351.33, binary search:27.40

testfile\_x10.txt

serial search:701.66, binary search: 34.70

testfile\_x1000.txt

serial search:70,066, binary search:72.23

Η καλύτερη μέθοδος αναζήτησης είναι ξεκάθαρα η δυαδική αναζήτηση, αφού ακόμα και για μικρά αρχεία είναι αρκετά πιο γρήγορη. Ο μέσος αριθμός προσβάσεων για ανεπιτυχή αναζήτηση μεταβάλλεται λογαριθμικά(αφού η binary search έχει πολυπλοκότητα  $O(\log n)$ ).

Πηγές:

cmd argument

<https://www.cs.colostate.edu/helpdocs/eclipseCommLineArgs.html>

Μηδενισμός ενός byte array(χρησιμοποιήθηκε το τελευταίο σχόλιο)

<https://stackoverflow.com/questions/44455987/resetting-byte-buffer-to-zeros>

αποθήκευση αρχείου

<https://stackoverflow.com/questions/5797208/java-how-do-i-write-a-file-to-a-specified-directory>

Επίσης χρησιμοποιήθηκαν και οι σημειώσεις του μαθήματος ΠΛΗ 102 για την δημιουργία της διπλά συνδεδεμένης λίστας, αλλά και οι διαφάνειες του μαθήματος ΠΛΗ 202 για την υλοποίηση της δυαδικής αναζήτησης.

Δεν βρέθηκε κάποιο πρόβλημα στην λειτουργία του προγράμματος.