ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΑΛΓΟΡΙΘΜΟΙ

3η άσκηση

Αλλαγές σε σχέση με την αρχική εκφώνηση:

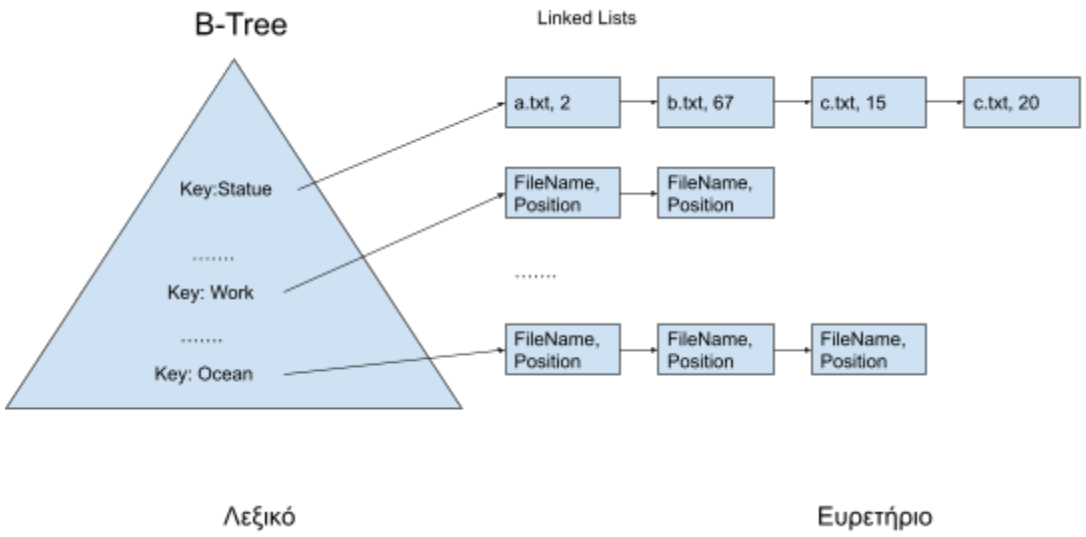
- 8/5/2023 αντί χρήση B-Tree, ζητείται να χρησιμοποιήσετε B+ Tree
- 8/5/2023 δίνονται 2 αρχεία κειμένων που θα εισάγετε
- 8/5/2023 διευκρινίσεις για τις μετρήσεις που ζητούνται

Ημερομηνία παράδοσης: 2 Ιουνίου 2023 ή ημερομηνία λήξης μαθημάτων χωρίς καμία παράταση.

<u>Αναζήτηση σε Αρχεία κειμένου με χρήση Δεικτοδότησης (indexing)</u>

Κατασκευάστε ένα πρόγραμμα που διαβάζει ένα ή περισσότερα αρχεία κειμένου (όπως αυτά που σας δίνονται ενδεικτικά) και δημιουργεί μια δομή δεδομένων στη μνήμη που απαντά σε ερωτήσεις όπως «**βρες κείμενα που περιέχουν την λέξη "Statue"** ή οποιαδήποτε άλλη λέξη. Τα αρχεία περιέχουν κείμενο που είναι ολόκληρο σε μία γραμμή και αποτελείται μόνο από λατινικούς ASCII χαρακτήρες. Η αναζήτηση γίνεται με την βοήθεια «δεικτοδότησης» (indexing) δηλαδή όχι απευθείας πάνω στα κείμενα αλλά μέσω μιας δομής στη μνήμη που σκοπό έχει να επιταχύνει την αναζήτηση στην περίπτωση που τα κείμενα είναι πάρα πολλά. Η δομή αποτελείται από το «**Λεξικό**» και το «**Ευρετήριο**». Η δομή στο σχήμα δηλώνει ότι η λέξη "Statue" του Λεξικού υπάρχει στην πρώτη λίστα του ευρετηρίου. Εκεί βρίσκεται η πληροφορία ότι η λέξη υπάρχει στα αρχεία a.txt, b.txt, c.txt στις θέσεις 2, 67, 15 και 20.

Για το λεξικό θα χρησιμοποιηθεί ένα B+-Tree και για το Ευρετήριο μία συνδεδεμένη λίστα. Κάθε λέξη αρχείου είναι ένα κλειδί. Κάθε κλειδί (λέξη) μπορεί να υπάρχει σε ένα ή περισσότερα αρχεία και σε μία ή περισσότερες θέσεις ενός αρχείου. Αυτή η πληροφορία αποθηκεύεται στο Ευρετήριο.



Θα χρησιμοποιήσετε ένα B+-tree τάξης M. Δηλαδή ότι όταν ο κόμβος (είτε εσωτερικός, είτε φύλλο) φτάσει να έχει M κλειδιά, προκαλείται διάσπαση. Θα πάρετε έτοιμες υλοποιήσεις από το Web (όποιες θέλετε). Δεν έχει σημασία η γλώσσα προγραμματισμού. **Για παράδειγμα**

B+-tree:
https://github.com/linli2016/BPlusTree

Ανάλυση Απόδοσης και Τεκμηρίωση

Γράψτε ένα πρόγραμμα με 3 λειτουργίες:

Α) Ρωτάει για όνομα αρχείου κειμένου (1 γραμμή, λατινικοί χαρακτήρες ASCII) και εισάγει την πληροφορία των λέξεων στη δομή που περιγράφεται παραπάνω. Δηλαδή, για κάθε λέξη του αρχείου,

- αν η λέξη βρίσκεται ήδη στη δομή, προσθέτει στην αντίστοιχη λίστα του ευρετηρίου την πληροφορία θέσης της λέξης και το όνομα αρχείου.
- αν δεν υπάρχει στη δομή, δημιουργεί τη λίστα του ευρετηρίου, και εισάγει στο B+Tree τη νέα λέξη μαζί με το δείκτη προς τη νέα λίστα του ευρετηρίου.

Β) Ρωτάει για μία λέξη προς αναζήτηση και τυπώνει την πληροφορία που υπάρχει για αυτήν (σε ποια αρχεία και σε ποια θέση βρίσκεται η λέξη στα διάφορα αρχεία που έχουν εισαχθεί). Ταυτόχρονα, κατά την αναζήτηση, υπολογίζει το πλήθος κόμβων του B+Tree που έκανε προσπέλαση και τον αριθμό συγκρίσεων που έγιναν με κλειδιά του δέντρου.

Γ) Κάνει 100 αναζητήσεις για λέξεις που υπάρχουν στο κείμενο για τη συμπλήρωση του παρακάτω πίνακα.

Χρησιμοποιήστε ως κείμενα εισόδου τα αρχεία 1.txt και 2.txt που σας δίνονται μαζί με την εκφώνηση και εισάγετέ τα στο δέντρο.

Από τα 2 αρχεία κειμένου, επιλέξτε τυχαία 50 λέξεις από το πρώτο και 50 από το δεύτερο, και κάντε αναζήτηση για αυτές για να εξάγετε τις μετρήσεις που ζητούνται στον παρακάτω πίνακα.

A1. Μέσος αριθμός προσβάσεων κόμβων στο Ευρετήριο (B+-tree) για αναζήτηση Τάξη M=10	A2. Μέσος αριθμός συγκρίσεων με κλειδιά για αναζήτηση Τάξη M=10	B1. Μέσος αριθμός προσβάσεων κόμβων στο Ευρετήριο (B+-tree) για αναζήτηση Τάξη M=20	B2. Μέσος αριθμός συγκρίσεων με κλειδιά για αναζήτηση Τάξη M=20

Σχολιάστε τα αποτελέσματα και αιτιολογήσετε τις μετρήσεις.

Πώς θα αξιολογούσατε τα αποτελέσματα εάν το B+-Tree και το Ευρετήριο ήταν στο δίσκο;

Παραδοτέα: Ένα συμπιεσμένο zip αρχείο που περιέχει τον πηγαίο κώδικα (φάκελος src) και την ζητούμενη αναφορά (όχι zip μέσα στο zip!).

Γενικοί κανόνες για τον κώδικα και την αναφορά:

- Ο κώδικας περιέχει συνοπτικά σχόλια που εξηγούν την υλοποίηση.
- Μία έκθεση που περιγράφει σε 1-2 σελίδες πως φτιάχτηκε ο κώδικας (δηλ. για κάθε ερώτημα ποια είναι η γενική ιδέα της λύσης σε 3-4 προτάσεις), υπάρχουν σαφείς οδηγίες μετάφρασης από compiler και εκτέλεσης, τι λάθη έχει (αν έχει, περιπτώσεις που δεν δουλεύει το πρόγραμμα, ή περιπτώσεις που κάνει περισσότερα από όσα σας ζητεί η άσκηση, τι χρησιμοποιήσατε από έτοιμα προγράμματα ή πηγές πληροφόρησης. Υποδείξετε ακόμα και πηγές στο WWW όπως Wikipedia.
- Οι ασκήσεις υποβάλλονται ηλεκτρονικά στον ιστοχώρο του μαθήματος και όχι με e-mail
- Οι αντιγραφές μηδενίζονται.
- Δώστε μεγάλη σημασία στην ποιότητα του κώδικα που θα στείλετε. Θα πρέπει να ακολουθεί τουλάχιστον τους παρακάτω κανόνες:
    - Χρησιμοποιείστε εύστοχα ονόματα μεταβλητών και μεθόδων έτσι ώστε να γίνεται εύκολα αντιληπτός ο λόγος και ο τρόπος χρήσης τους.

- Γράψτε απλό και δομημένο κώδικα ώστε αυτός που τον διαβάζει να μπορεί να καταλάβει τα βήματα που ακολουθούνται για την υλοποίηση του κάθε προβλήματος.
- Βάλτε εύστοχα σχόλια όπου χρειάζεται.
- Κάντε το κώδικά σας παραμετρικό με χρήση μεταβλητών μέσω των οποίων θα μπορείτε να αλλάζετε κάποια μεγέθη (π.χ. τα μεγέθη για N, M κ.ο.κ.) χωρίς να χρειάζεται να αλλάζετε κάθε φορά και την αντίστοιχη υλοποίηση.