

Δομές Δεδομένων

1^η Εργασία

2022-2023

Στα πλαίσια της εργασίας ζητείται να συγγράψετε πρόγραμμα σε C++ το οποίο θα επιλύει το πρόβλημα που ακολουθεί.

Δίνεται ένα σύνολο από 100 αρχεία, με ονόματα 1.txt, 2.txt, 3.txt, ..., 99.txt, 100.txt. Σκοπός της εργασίας είναι να δημιουργήσετε ένα σύνολο από γραμμικές λίστες, οι οποίες θα διατηρούν πληροφορία σχετικά με τις λέξεις που περιέχουν τα αρχεία αυτά, όπως περιγράφεται παρακάτω. **Στα πλαίσια της εργασίας θα πρέπει να χρησιμοποιήσετε την συνδεδεμένη αναπαράσταση για τις γραμμικές λίστες.** Για λεπτομέρειες δείτε το Κεφ. 3.4 του βιβλίου “Δομές Δεδομένων, Αλγόριθμοι και Εφαρμογές στη C++”. **Δεν επιτρέπεται η χρήση των έτοιμων δομών δεδομένων που προσφέρει η C++ (C++ Containers), ούτε η χρήση πινάκων για την πραγματοποίηση των λειτουργιών που περιγράφονται παρακάτω και η μετατροπή στην συνέχεια σε λίστα.**

Σκοπός του προγράμματος που θα φτιάξετε είναι να δημιουργήσετε μια «λίστα από λίστες». Συγκεκριμένα, το πρόγραμμα θα διαβάζει μια προς μια τις λέξεις που υπάρχουν σε κάθε αρχείο εισόδου. Κάθε λέξη θα αποθηκεύεται σε έναν κόμβο της λίστας LOW (List Of Words). Οι κόμβοι της λίστας LOW θα αποτελούνται από τα εξής δεδομένα:

- Ένα αλφαριθμητικό (string), στο οποίο θα αποθηκεύεται η λέξη.
- Έναν ακέραιο αριθμό, στον οποίο θα αποθηκεύεται το πλήθος των αρχείων στα οποία βρέθηκε η λέξη.
- Έναν δείκτη προς μια λίστα LOF (List Of Files), στην οποία θα αποθηκεύονται οι αριθμοί των αρχείων στα οποία βρέθηκε η συγκεκριμένη λέξη (περισσότερες λεπτομέρειες στην συνέχεια).
- Έναν δείκτη προς το επόμενο στοιχείο της λίστας LOW.

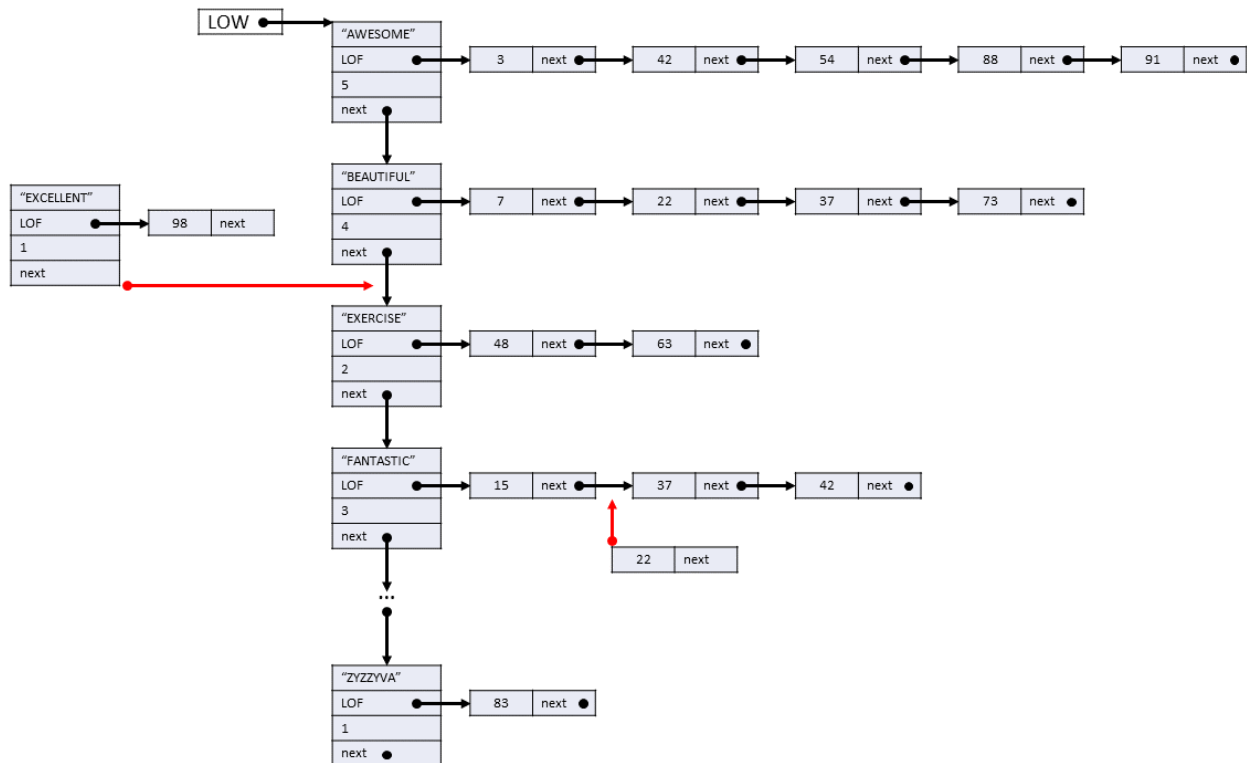
Για κάθε λέξη που διαβάζει το πρόγραμμα θα ελέγχει αν υπάρχει ήδη στην λίστα LOW.

- Αν δεν υπάρχει, θα πρέπει να δημιουργεί έναν νέο κόμβο και να τον τοποθετεί στο σωστό σημείο της λίστας LOW, ώστε αυτή να διατηρείται ταξινομημένη. Η εισαγωγή στην σωστή θέση θα πρέπει να γίνεται μόλις δημιουργηθεί ο νέος κόμβος. Δεν επιτρέπεται η δημιουργία όλης της λίστας LOW και η ταξινόμηση της στο τέλος.
- Αν υπάρχει, τότε θα πρέπει να ελέγχει αν ο αριθμός του αντίστοιχου αρχείου υπάρχει ήδη στην λίστα LOF της συγκεκριμένης λέξης.
 - Αν υπάρχει, δεν χρειάζονται περαιτέρω ενέργειες.
 - Αν δεν υπάρχει, τότε θα πρέπει να δημιουργεί έναν νέο κόμβο και να τον τοποθετεί στο σωστό σημείο της λίστας LOF, ώστε αυτή να διατηρείται ταξινομημένη. Η εισαγωγή στην σωστή θέση θα πρέπει να γίνεται μόλις δημιουργηθεί ο νέος κόμβος. Δεν επιτρέπεται η δημιουργία όλης της λίστας LOF και η ταξινόμηση της στο τέλος. Οι κόμβοι κάθε λίστας LOF θα αποτελούνται από τα εξής δεδομένα:

- Έναν ακέραιο αριθμό, στον οποίο θα αποθηκεύεται ο αριθμός του αρχείου στο οποίο βρέθηκε η λέξη.
- Έναν δείκτη προς το επόμενο στοιχείο της λίστας LOF.

Το παρακάτω σχήμα δίνεται προς διευκρίνιση λεπτομερειών. Για παράδειγμα, υποθέστε πως κατά την διάρκεια επεξεργασίας του αρχείου 98.txt βρέθηκε η λέξη “EXCELLENT”. Η λέξη αυτή στο παράδειγμα μας δεν έχει βρεθεί σε άλλο αρχείο νωρίτερα. Αφού κατασκευαστεί ο αντίστοιχος κόμβος για την λίστα LOW, αυτός θα πρέπει να τοποθετηθεί μετά την λέξη “BEAUTIFUL” και πριν την λέξη “EXCELLENT” (όπως δείχνει το κόκκινο βέλος), ώστε η λίστα LOW να παραμείνει ταξινομημένη. Βεβαίως θα πρέπει να δημιουργηθεί και ο αντίστοιχος κόμβος της λίστας LOF για την λέξη “EXCELLENT”, ώστε να δηλωθεί πως η λέξη αυτή έχει βρεθεί σε ένα αρχείο συνολικά, το οποίο είναι το αρχείο 98.txt.

Υποθέστε τώρα πως κατά την επεξεργασία του αρχείου 22.txt βρέθηκε η λέξη “FANTASTIC”. Η λέξη αυτή υπάρχει ήδη στην λίστα LOW, επομένως δεν χρειάζεται να προστεθεί νέος κόμβος στην λίστα αυτή. Θα πρέπει όμως να δημιουργηθεί νέος κόμβος για την λίστα LOF της λέξης αυτής και να τοποθετηθεί μεταξύ του 15 και του 37 (όπως δείχνει το κόκκινο βέλος), ώστε η λίστα LOF της λέξης “FANTASTIC” να παραμείνει ταξινομημένη. Βεβαίως θα πρέπει να ενημερωθεί και ο μετρητής και από 3 να γίνει 4, ώστε να δηλώνεται πως η λέξη αυτή έχει βρεθεί συνολικά σε 4 αρχεία. Τέλος, αν ξαναβρεθεί η ίδια λέξη στο ίδιο αρχείο δεν χρειάζεται να γίνει καμία αλλαγή στις λίστες.



Μετά την ολοκλήρωση της δημιουργίας των λιστών, το πρόγραμμα θα εισέρχεται σε έναν βρόχο (loop), στον οποίο θα ζητάει από τον χρήστη να εισάγει μια λέξη. Στην συνέχεια θα ελέγχει αν η λέξη αυτή υπάρχει στην λίστα LOW. Αν υπάρχει, θα τυπώνει σε πόσα αρχεία έχει βρεθεί η λέξη και σε ποια (τους αριθμούς των αρχείων δηλαδή). Αν δεν υπάρχει, θα τυπώνει ένα αντίστοιχο μήνυμα. Χρησιμοποιώντας το παράδειγμα της εικόνας, αν ο χρήστης δώσει ως είσοδο την λέξη “BEAUTIFUL” θα πρέπει να τυπωθεί ένα μήνυμα της μορφής:

The word “BEAUTIFUL” appears in 4 documents. These documents are:
7, 22, 37, 73

Το πρόγραμμα θα βγαίνει από αυτόν τον βρόχο και θα τερματίζει όταν ο χρήστης δώσει τον ειδικό χαρακτήρα End-Of-File (EOF).

Θέματα υλοποίησης

- Η σειρά επεξεργασίας των αρχείων **δεν** θα γίνει με βάση την αριθμητική τιμή που περιέχεται στην ονομασία τους (δεν θα γίνει δηλαδή με την σειρά 1.txt, 2.txt, 3.txt, ..., 99.txt, 100.txt). Χρησιμοποιήστε τον πίνακα “orderOfFiles” που δίνεται στο συνοδευτικό αρχείο “orderOfFiles.cpp” και επεξεργαστείτε τα αρχεία με την σειρά αυτή. Μπορείτε να αντιγράψετε τον ορισμό του πίνακα στο πρόγραμμα σας.
- Στα αρχεία που δίνονται έχει πραγματοποιηθεί κάποια προεπεξεργασία. Έχουν αφαιρεθεί σημεία στίξης (τελεία, κόμμα, θαυμαστικό, ερωτηματικό, κλπ) και έχουν μετατραπεί όλοι οι χαρακτήρες σε κεφαλαία. **Θεωρείστε πως μια λέξη αποτελείται από τους χαρακτήρες μεταξύ δύο κενών χαρακτήρων.** Η προσέγγιση αυτή απλοποιεί την επεξεργασία που απαιτείται από την μεριά σας. Αγνοήστε τυχόν άλλα προβλήματα που μπορεί να προκύψουν από την προσέγγιση αυτή (π.χ. ότι το “IT’S” λόγω αντικατάστασης της αποστρόφου με κενό κατά την προεπεξεργασία θα θεωρηθεί τελικά δύο λέξεις (“IT” και “S”).

Παραδοτέα

Θα πρέπει να παραδοθεί ο πηγαίος κώδικας μαζί με τον εκτελέσιμο. Ιδιαίτερη βαρύτητα θα πρέπει να δοθεί στη σωστή τεκμηρίωση των προγραμμάτων σας. Θα πρέπει λοιπόν ο κώδικας σας να συνοδεύεται από ξεχωριστό κείμενο που θα παρέχει λεπτομερή περιγραφή των τεχνικών σας. Επίσης, εντός του πηγαίου κώδικα θα πρέπει να συμπεριληφθούν «πυκνά» **σχόλια ουσίας**. Η παράδοση των εργασιών θα γίνει μέσω του gunet.

Η εργασία μπορεί να εκπονηθεί από **ομάδα μέχρι δύο ατόμων αυστηρώς**.

Προθεσμία Παράδοσης

Τρίτη 2 Μαΐου 2023