

Γενικά

Το πρόγραμμα υλοποιεί όλα όσα έχουν ζητηθεί και περιλαμβάνει και μια υλοποίηση για το `backspace`. Σαν `input` δέχεται όλους τους χαρακτήρες, αλλά στο δέντρο, όπως και στο λεξικό κατά την έξοδο, εισάγει μόνο τις έγκυρες λέξεις.

Σχεδιαστικές Επιλογές

Υπάρχουν 2 κλάσεις, η μια εκ των οποίων περιέχει αρκετές υποκλάσεις και `structs`. Η πρώτη κλάση είναι μια λίστα από `strings`, η οποία χρησιμοποιείται για τις λίστες με τις λέξεις με τις μεγαλύτερες συχνότητες.

Η δεύτερη κλάση είναι το `trie`. Το `trie` περιλαμβάνει έναν δείκτη στο `node root` και έναν δείκτη στο `node current`, 2 μεταβλητές `bool` για το αν η τρέχουσα λέξη είναι όντως λέξη και αν είναι έγκυρη, καθώς και μια λίστα με τις νέες λέξεις.

(Σημ: Η πρώτη κλάση θα μπορούσε να παραληφθεί, γιατί άλλαξα μια σχεδιαστική επιλογή, αλλά δεν είχα χρόνο να το αλλάξω και να βεβαιωθώ ότι δεν έχει προβλήματα.

Το `node` αποτελείται από μια λίστα με δείκτες σε άλλα `node`, από μια λίστα με τα φύλλα με την μεγαλύτερη συχνότητα, έναν δείκτη στον γονιό του και έναν δείκτη στο φύλλο της λέξης που αντιπροσωπεύει.

Σχεδιαστικές Επιλογές

Το πρόγραμμα δέχεται όλους τους χαρακτήρες. Στο δέντρο εισάγονται μόνο οι έγκυρες λέξεις. Μια λέξη είναι έγκυρη όταν αρχίζει στην αρχή μιας γραμμής ή μετά από κενό, και τελειώνει σε κενό, αλλαγή γραμμής ή κάποιο σημείο στίξης από τα παρακάτω: `.,;*/?!-\\:`

Σε άλλη περίπτωση, δεν αποθηκεύεται στο δέντρο.

Χρήση Tab

Όταν ο χρήστης πατήσει το `tab`, μπορεί να συμβεί μια από της παρακάτω περιπτώσεις.

- 1) Ο χρήστης βρίσκεται σε σημείο του δέντρου που δεν υπάρχουν λέξεις. Ο χρήστης τότε ειδοποιείται για αυτό και συνεχίζει την εισαγωγή.
- 2) Ο χρήστης βρίσκεται σε σημείο του δέντρου που υπάρχει μια μόνο λέξη. Ο χρήστης τότε ειδοποιείται για αυτό και η λέξη συμπληρώνεται αυτόματα.
- 3) Ο χρήστης βρίσκεται σε σημείο με πολλές λέξεις. Ο χρήστης βλέπει τις επιλογές του και διαλέγει την λέξη που θέλει. Υπάρχουν και αριθμοί για ευκολότερη επιλογή. Υποστηρίζονται μέχρι δέκα πιθανές επιλογές. Η πρώτη επιλογή γίνεται πάντα με τον αριθμό 1, και σε περίπτωση που το `N` είναι 10, η τελευταία επιλογή είναι το 0. Στην περίπτωση που ο χρήστης είναι στο `root`, παίρνει τις `N` λέξεις με μεγαλύτερη συχνότητα ανεξαρτήτως προθέματος.
- 4) Ο χρήστης δεν βρίσκεται σε μέσα σε μια έγκυρη λέξη (πχ στην περίπτωση `a4v`). Τότε ο χρήστης ενημερώνεται ότι δεν βρίσκεται σε έγκυρη λέξη και συνεχίζει την εισαγωγή.

Επίσης πρέπει να σημειωθεί ότι το `tab` δεν εισάγει λέξεις ή αυξάνει συχνότητα. Αυτό θα συμβεί μόλις ο χρήστης εισάγει τον επόμενο χαρακτήρα, αν αυτός τερματίζει την λέξη. Αυτό συμβαίνει για να δοθεί η επιλογή στον χρήστη να εισάγει περισσότερα γράμματα στην λέξη, καθώς και να την τερματίσει όπως αυτός επιθυμεί (πχ με τελεία αντί για κενό).

Σε περίπτωση που ο χρήστης έχει επιλογές και πατήσει το tab, επιστρέφει στο σημείο που ήταν. Αν πατήσει χαρακτήρα εκτός των επιλογών του, το πρόγραμμα συνεχίζει σαν να μην είχε πατηθεί το tab.

Περιορισμοί Backspace

Το πρόγραμμα υποστηρίζει και backspace. Οι μόνοι περιορισμοί που έχει αυτό είναι ότι αν έχει εισαχθεί ήδη μια λέξη, τότε αυτή δεν μειώνεται η συχνότητα της, ούτε βγαίνει από τις λίστες.

Μερικά παραδείγματα backspace.

1) Abc.

Με το που θα πατήσουμε το backspace, το δέντρο θα ξαναβρεθεί στο σημείο του προθέματος abc, και θα βρίσκεται μέσα σε έγκυρη λέξη. Το abc όμως εξακολουθεί να έχει την συχνότητα που είχε αφού το πληκτρολογήσαμε.

2) abc3

Όταν σβήσουμε το 3, θα είμαστε στο σημείο abc, μέσα σε έγκυρη λέξη.

3) abc def

Αν σβήσουμε 4 γράμματα, θα είμαστε στο σημείο abc του δέντρου, σε έγκυρη λέξη.

Ctrl + D

Όταν εισάγει αυτόν τον συνδυασμό ο χρήστης, το πρόγραμμα τερματίζεται. Αν βρίσκεται σε έγκυρη λέξη, αυτή εισάγεται στο δέντρο.