



ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΕΠΑ 035 - ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΑΛΓΟΡΙΘΜΟΙ ΓΙΑ ΗΛΕΚΤΡΟΛΟΓΟΥΣ
ΜΗΧΑΝΙΚΟΥΣ ΚΑΙ ΜΗΧΑΝΙΚΟΥΣ ΥΠΟΛΟΓΙΣΤΩΝ
Ακαδημαϊκό έτος 2020-2021

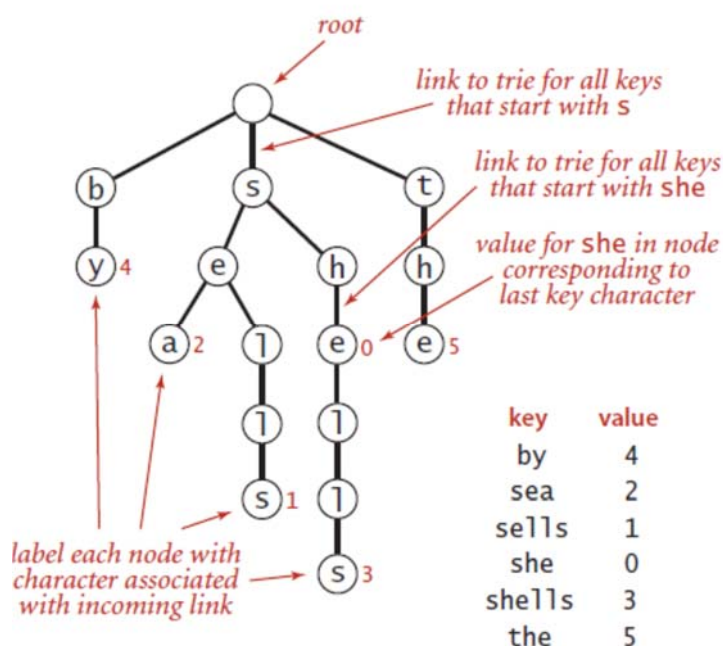
Υπεύθυνος εργαστηρίου: Πέτρος Παναγή

Εργασία 2 – Δεντρικές Δομές Δεδομένων

Οι περισσότερες εταιρίες που εμπορεύονται αγαθά και παρέχουν υπηρεσίες διαδικτυακά δίνουν την δυνατότητα αγοράς και χρήσης κουπονιών μίας χρήσης (gift cards, codes κτλ.). Για παράδειγμα, αν θες να προσθέσεις χρόνο ομιλίας στο κινητό σου, μπορείς να πληρώσεις από της ιστοσελίδα του παρόχου ή σε ένα κατάστημα, το ποσό που θες να πιστώσεις στο κινητό σου, ο πάροχος θα σου δώσει ένα 16ψήφιο μυστικό αριθμό πχ. 1234567890123456 τον οποίο αφού καλέσεις την ανάλογη υπηρεσία και καταχωρίσεις τον κωδικό, ο λογαριασμός του κινητού σου θα πιστωθεί με το ανάλογο ποσό.

Ο κωδικός που σου έχει δοθεί είναι μίας χρήσης και δεν μπορεί πλέον να χρησιμοποιηθεί. Επιπρόσθετα ο κωδικός έχει δημιουργηθεί με μία τυχαία σειρά αριθμών ώστε να μην μπορεί κάποιος να τον παράξει χρησιμοποιώντας μία συγκεκριμένη διαδικασία. Με άλλα λόγια ο κωδικός που σας έχει δοθεί είναι μοναδικός, μη προβλέψιμος και μίας χρήσης.

Σε αυτή την εργασία σας ζητείτε να υλοποιήσετε ένα τέλειο σύστημα με την βοήθεια Trie. Τα Trie είναι δένδρα αναζήτησης των οποίων ο κάθε κόμβος αποτελείται από πολλούς κλάδους (παιδιά). Για παράδειγμα, ο κάθε κλάδος παιδί αντιπροσωπεύει έναν χαρακτήρα του αλφαβήτου όπως φαίνεται στο πιο κάτω σχήμα.



Αν για παράδειγμα το λεξικό που θέλουμε να αναπαραστήσουμε σε ένα Tries περιέχει την λέξη Shells, θα πρέπει να εισάγουμε 6 κόμβους. Επίσης μία λέξη μπορεί να είναι μέρος μιας μεγαλύτερης λέξης, όπως

στο πιο πάνω παράδειγμα το she που είναι μέρος του **shells**. Άρα θα πρέπει να επισημάνουμε τον τελευταίο κόμβο κάθε λέξης ως «κόμβο τέλους λέξης» π.χ. με ένα πεδίο **isTheEndOfKey**. Τέλος στην πιο πάνω υλοποίηση θα πρέπει να γνωρίζουμε αν ο κωδικός έχει χρησιμοποιηθεί. Αυτό μπορεί να γίνει με την βοήθεια ενός πεδίου π.χ. **isUsed**. Η υλοποίηση ενός κόμβου Trie για την άσκηση μας θα πρέπει να είναι ο πιο κάτω:

```
struct TrieNode
{
    struct TrieNode *letter[26];
    bool isTheEndOfKey;
    bool isUsed;
};
```

Λεπτομέρειες Υλοποίησης

1. Το πρόγραμμά σας θα παίρνει ως παράμετρο από τη γραμμή εντολής το όνομα του αρχείου που περιέχει τους κωδικούς μίας χρήσης που θα περαστούν στο σύστημα (έστω input.txt). Οι κωδικοί θα αποτελούνε από 10 κεφαλαία γράμματα του αγγλικού αλφαβήτου. Π.χ.

```
ABCDEFGHIJ
JIHGFEDCBA
AAAAAAAAAA
BBBBBBBBBB
ZZZZZZZZZZ
```

2. Θα διαβάσει το αρχείο input.txt και θα δημιουργεί το αντίστοιχο δένδρο.
3. Θα δίνει την δυνατότητα αναζήτησης του κωδικού μέσω πληκτρολογίου. Ο κωδικός θα μπορεί να επαληθευτεί ΜΟΝΟ μία φορά. Οπότε αν γίνει αναζήτηση του κωδικού ταυτόχρονα ο κωδικός αυτός απενεργοποιείται. Π.χ.

```
AAAAAAAAAA --- Code is Valid
ABCDEFGHIJ --- Code is Valid
FFFFFFFFFF --- Code is NOT Valid
GGGGGGGGGG --- Code is NOT Valid
AAAAAAAAAA --- Code is NOT Valid
```

Αξιολόγηση Εργασίας

Η προγραμματιστική αυτή άσκηση είναι ατομική και η μέγιστη δυνατή βαθμολογία είναι το 100. Για την αξιολόγηση της προγραμματιστικής αυτής άσκησης, θα ληφθούν οι πιο κάτω παράμετροι:

1. **Δομή Συστήματος 5%:** Το πρόγραμμά σας θα πρέπει να χρησιμοποιεί τεχνικές δομημένου προγραμματισμού με τη χρήση συναρτήσεων, αρχείων επικεφαλίδας (.h) και πολλαπλών αρχείων για καλύτερη δομή του πηγαίου κώδικα και διαχείριση λαθών.
2. **Ορθότητα Λειτουργίας 40%:** Το πρόγραμμά σας θα πρέπει να υλοποιεί ορθά τις λειτουργίες του συστήματος όπως αυτές περιγράφονται στην εκφώνηση. Ένα πρόγραμμα που δεν δίνει τη σωστή απάντηση δεν έχει καμιά αξία. Τα προγράμματα θα πρέπει να δίνουν τη σωστή απάντηση για όλα τα στιγμιότυπα του πεδίου ορισμού του προβλήματος που λύνουν, διαφορετικά δεν θα λάβουν (όλες) τις μονάδες που αντιστοιχούν στην ορθότητα. Βεβαιωθείτε ότι έχετε ελέγξει την ορθότητα του προγράμματός σας στις μηχανές του εργαστηρίου ΘΕΕ01-B103 πριν το παραδώσετε.
3. **Επίδοση/Επεξήγηση Συστήματος 25%:** Το πρόγραμμά σας θα πρέπει να ελαχιστοποιεί τη χρήση των πόρων του συστήματος και πιο συγκεκριμένα της μνήμης και του επεξεργαστή. Ο

αλγόριθμος που χρησιμοποιείτε θα πρέπει να περιγράφεται με σαφήνεια και ο κώδικας πρέπει να είναι καθαρά γραμμένος και να περιέχει επεξηγηματικά σχόλια.

4. **Περιγραφή λύσης και γραφική αναπαράσταση δένδρου (30%).** Να δώσετε πλήρη περιγραφή της υλοποίησης σας δικαιολογώντας τις αποφάσεις σας. Επίσης σε μία σελίδα A4 να αναπαραστήσετε τον δένδρο για τις τιμές AAAA, AAAB, AABA, ABAA.
5. Σε περίπτωση που το πρόγραμμά σας δεν τρέχει στις μηχανές της ΘΕΕ01-B103 ή δεν βγάζει αποτέλεσμα ή έχει segmentation fault η μέγιστη βαθμολογία της εργασίας σας θα είναι 30.

Οδηγίες Υποβολής

Η εργασία θα παραδοθεί μόνο ηλεκτρονικά σε ένα αρχείο 123456.zip όπου το 123456 είναι ο αριθμός ταυτότητας σας.

Η παράδοση θα γίνει μέσω του BlackBoard (EPL035_Labs). **Προθεσμία: 22/11@23:59**

Δημιουργήστε ένα φάκελο στον υπολογιστή σας που θα έχει ως όνομα **μόνο την ταυτότητά σας**. Τοποθετήστε σε αυτό τα ακόλουθα αρχεία:

- a. Τα αρχεία με τον πηγαίο κώδικα του προγράμματός σας
- b. Ένα αρχείο readme.txt το οποίο θα περιέχει τις οδηγίες χρήσης του προγράμματός σας
- c. Ένα αρχείο implementation.pdf το οποίο θα περιγράφει τις συναρτήσεις που χρησιμοποιήσατε καθώς και το δένδρο στο ερώτημα 3 πιο πάνω (2 σελίδες)

ΚΑΛΗ ΕΠΙΤΥΧΙΑ