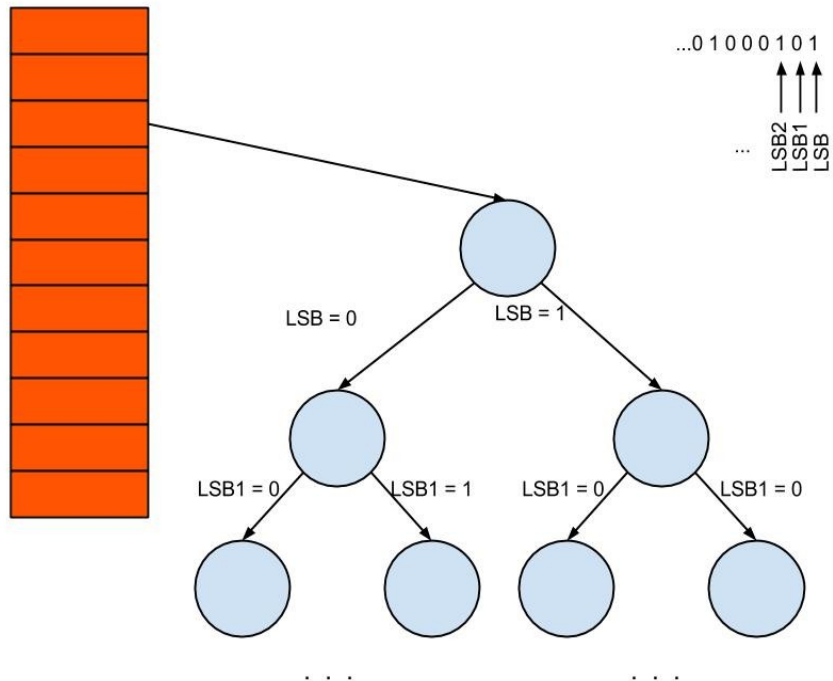


Δομές Δεδομένων & Αρχείων

Δυναμικός κατακερματισμός (dynamic hashing) στη μνήμη

Γενικά

Σκοπός της εργασίας αυτής ήταν η υλοποίηση δυναμικού κατακερματισμού στη μνήμη, με πράξεις αναζήτησης, εισαγωγής και διαγραφής στοιχείων. Η γενική ιδέα του κατακερματισμού θα μπορούσαμε να πούμε ότι είναι η απευθείας απεικόνιση στοιχείων-κλειδιών στις θέσεις ενός πίνακα, με τη χρήση κάποιας κατάλληλης συνάρτησης. Το πρόβλημα δημιουργείται όταν παραπάνω από ένα στοιχεία απεικονίζονται στην ίδια θέση του πίνακα. Υπάρχουν διάφορες λύσεις, που προσφέρουν ποικίλες πολυπλοκότητες, αλλά ο δυναμικός κατακερματισμός προσπαθεί να επιτύχει πολυπλοκότητα σταθερού όρου. Η υλοποίηση στην παρούσα εργασία έγινε με την χρήση μιάς δομής, που μοιάζει σε δυαδικό δέντρο και ονομάζεται trie, αλλά διαφέρει στον τρόπο με τον οποίο ταξινομούνται τα διάφορα κλειδιά. Στο δυαδικό δέντρο συγκρίνουμε το κλειδί που θέλουμε να εισάγουμε με τα κλειδιά που υπάρχουν ήδη, έτσι προχωράμε αριστερά ή δεξιά προς τα κάτω ανάλογα με το αποτέλεσμα της σύγκρισης. Στο trie κάνουμε χρήση της δυαδικής αναπαράστασης του αριθμού που θέλουμε να εισάγουμε και ελέγχουμε τα bits ανάλογα με το βάθος στο οποίο βρισκόμαστε. Στην εικόνα φαίνεται ένα σχέδιο της δομής αυτή, που βοηθάει αρκετά.



Υλοποίηση

Η υλοποίηση της εργασίας πραγματοποιήθηκε στη γλώσσα προγραμματισμού java, στο περιβάλλον του Eclipse. Δημιουργήθηκαν τέσσερεις κλάσεις.

- **TrieNode**
Αυτή η κλάση αναφέρεται στους κόμβους της δενδρικής δομής. Περιλαμβάνει τις μεταβλητές του κάθε κόμβου, καθώς επίσης και συναρτήσεις που δίνουν στοιχεία για το είδος του κόμβου, την κατάσταση του κτλ. Τέτοιες συναρτήσεις είναι οι isLeaf, isEmpty, isFull, print κλ.
- **Trie**
Αυτή η κλάση αναφέρεται στην δημιουργία της δενδρικής δομής, και περιλαμβάνει μεθόδους αναζήτησης,

διαγραφής, εισαγωγής, διάσπασης. Για παράδειγμα η μέθοδος `insertKeyToTrie` εισάγει νέο κλειδί στη δομή, χρησιμοποιώντας ως βοηθητική την `insertKeyToNode`, η οποία εισάγει ένα νέο κλειδί σε κόμβο που δεν είναι γεμάτος. Η `splitNode` πραγματοποιεί διάσπαση ενός κόμβου όταν αυτός είναι γεμάτος και ένα νέο κλειδί πρέπει να εισαχθεί κτλ.

- **HashTable**
Η κλάση αυτή δημιουργεί τον βασικό πίνακα του κατακερματισμού, του οποίου κάθε θέση δείχνει σε ρίζα μιας δενδρικής δομής όπως αυτή που περιγράψαμε παραπάνω. Εδώ περιέχεται και η βασική συνάρτηση κατακερματισμού, η οποία είναι $h = \text{key} \bmod 100$.
- **MyMain**
Σε αυτή την κλάση συνδυάζονται τα παραπάνω και παράγονται τα ζητούμενα αποτελέσματα.

Αποτελέσματα

Τα αποτελέσματα μετά από τυχαίες εκτελέσεις του προγράμματος μας φαίνονται δεξιά στις εικόνες. Αρχικά μπορούμε να παρατηρήσουμε ότι το μέσο βάθος του δέντρου είναι 2. Φαίνεται αρκετά λογικό αφού το εύρος των κλειδιών δεν περιορίζεται σε λίγους αριθμούς και επίσης κάθε κόμβος περιέχει 2 στοιχεία. Από αυτό και μόνο μπορούμε να συμπεράνουμε ότι μάλλον έχουμε μια καλή κατανομή των κλειδιών κατά μήκος του πίνακα κατακερματισμού.

Επίσης ο αριθμός των συγκρίσεων που γίνονται κατά την εισαγωγή είναι λογικός. Σίγουρα έχουμε μία σύγκριση για την εύρεση της θέσης του κλειδιού πάνω στον στατικό πίνακα. Επίσης αφού έχουμε μέσο βάθος δέντρου 2, σημαίνει ότι θα γίνουν μία με δύο συγκρίσεις ώστε να αποφασιστεί η σωστή θέση του κλειδιού.

```
Average Depth: 2
Average insert comparisons: 3
Average search comparisons: 2
Average delete comparisons: 1
```

Παρατηρούμε ότι σε ορισμένες περιπτώσεις όπως αυτή της πρώτης εικόνας ο αριθμός των συγκρίσεων κατά την αναζήτηση είναι μικρότερος από τον μέσο αριθμό συγκρίσεων κατά την εισαγωγή. Αυτό συμβαίνει διότι κατά την αναζήτηση μπορεί να ψάχνουμε κλειδιά που είναι σε μικρότερο βάθος.

```
Average Depth: 2
Average insert comparisons: 2
Average search comparisons: 2
Average delete comparisons: 1
```

Πηγές

<http://en.wikipedia.org/wiki/Trie>
<http://www.cs.sfu.ca/CourseCentral/354/zaiane/material/notes/Chapter11/node20.html>
<http://www.cs.bilkent.edu.tr/~canf/CS351Fall2009/cs351lecturenotes/week6/index.html>
https://www.google.gr/#sclient=psy-ab&q=dynamic+hashing+schemes&oq=dynamic+hashing+schemes&gs_l=serp.3..0i30.15094.16382.0.17806.7.3.0.4.4.0.109.305.1j2.3.0..0.0...1c.1.16.psy-ab.aV60WMJEJGc&pbx=1&bav=on.2.or.r_cp.r_qf.&fp=d09d317d819d35ed&biw=1855&bih=994

Σημειώσεις μαθήματος
Σύγγραμα: Δομές Δεδομένων του Π.Μποζάνη

Τέλος Αναφοράς