Δομές Δεδομένων

18η Διάλεξη

Ισορροπημένα δέντρα

Ε. Μαρκάκης

Περίληψη

- Επανάληψη των Τυχαιοποιημένων ΔΔΑ, Στρεβλών ΔΔΑ, Δέντρων 2-3-4
- Δέντρα κόκκινου-μαύρου
- Λίστες Παράλειψης
- Χαρακτηριστικά επιδόσεων - συμπεράσματα

Εισαγωγή

- Μειονεκτήματα δέντρων Δυαδικής Αναζήτησης (ΔΔΑ)
  - Κακή επίδοση χειρότερης περίπτωσης
    - Παράδειγμα: εισαγωγή ήδη ταξινομημένων κλειδιών
  - Λύση: διατήρηση των δέντρων σε ισορροπημένη μορφή
    - Όλοι οι εξωτερικοί κόμβοι στο ίδιο ή σε δύο διαδοχικά επίπεδα
  - Προτεινόμενες δομές:
    - Τυχαιοποιημένα ΔΔΑ
    - Στρεβλά ΔΔΑ
    - Δέντρα 2-3-4
    - Δέντρα κόκκινου-μαύρου
  - Πρόβλημα: διατήρηση ισορροπίας μετά από αλλαγές

Εισαγωγή

- Διατήρηση ΔΔΑ σε ισορροπημένη μορφή
  - Η περιοδική ανακατασκευή μπορεί να έχει μεγάλο κόστος
  - Χρειαζόμαστε μεθόδους τοπικής ανακατασκευής
- Προσεγγίσεις αποφυγής χειρότερης περίπτωσης
  - Τυχαιοποίηση (Randomization): στατιστική αποφυγή της χειρότερης περίπτωσης
    - Εισάγουμε κάποια τυχαία απόφαση στον αλγόριθμο
    - Παράδειγμα: τυχαιοποιημένα ΔΔΑ
  - Απόσβεση (Amortization): εκτέλεση επιπλέον εργασίας ορισμένες φορές
    - Το κόστος αποσβένεται μετά από πολλές λειτουργίες
    - Παράδειγμα: στρεβλά ΔΔΑ
  - Βελτιστοποίηση (optimization): εγγυήσεις επίδοσης σε κάθε λειτουργία
    - Χρήση πρόσθετων δομικών πληροφοριών μέσα στα ΔΔΑ
    - Παράδειγμα: δέντρα 2-3-4, δέντρα κόκκινου-μαύρου

Τυχαιοποιημένα ΔΔΑ

- Απλή παραλλαγή των ΔΔΑ
  - Έστω ένα ΔΔΑ με N κόμβους
  - Με πιθανότητα 1/(N+1), το επόμενο στοιχείο εισάγεται στη ρίζα
    - Αυτή είναι και η πιθανότητα να είναι η ρίζα ενός τυχαίου ΔΔΑ
  - Διαφορετικά αναδρομή στο αντίστοιχο υποδέντρο
- Επιδόσεις
  - Κατασκευή: περίπου 2NlnN συγκρίσεις
  - Αναζήτηση: περίπου 2lnN συγκρίσεις
  - Αμελητέα πιθανότητα το δένδρο να είναι μη ισορροπημένο
- Μειονεκτήματα τυχαιοποιημένων ΔΔΑ
  - Οι καλές γεννήτριες τυχαίων αριθμών είναι αργές
    - Χρήση γρήγορων αλλά όχι τόσο τυχαίων γεννητριών
  - Απαιτείται πεδίο πλήθους κόμβων σε κάθε κόμβο
    - Μπορεί όμως να χρησιμοποιείται και για άλλους λόγους

Στρεβλά ΔΔΑ (Splay Trees)

- Παραλλαγή των ΔΔΑ με εισαγωγή στη ρίζα
  - Λαμβάνουμε υπόψη δύο διαδοχικές περιστροφές
  - Αν είναι προς αντίθετες κατευθύνσεις, καμία αλλαγή
  - Αν είναι προς την ίδια κατεύθυνση, διαφοροποιούμαστε
    - Κάνουμε πρώτα την περιστροφή στη ρίζα και μετά στο παιδί
- Τι κερδίζουμε με τη στρεβλή εισαγωγή;
  - Σε κάθε εισαγωγή μικραίνει η διαδρομή εισαγωγής
    - Το μήκος της διαδρομής μειώνεται στο μισό (μπορεί βέβαια να αυξάνεται το μήκος άλλων διαδρομών)
- Βελτίωση απόδοσης στρεβλών ΔΔΑ
  - Μπορούμε να κάνουμε εξισορρόπηση και στις αναζητήσεις
  - Ιδιαίτερα αποδοτική για ανομοιόμορφες προσπελάσεις
    - Τα κλειδιά που χρησιμοποιούνται συχνά πλησιάζουν τη ρίζα

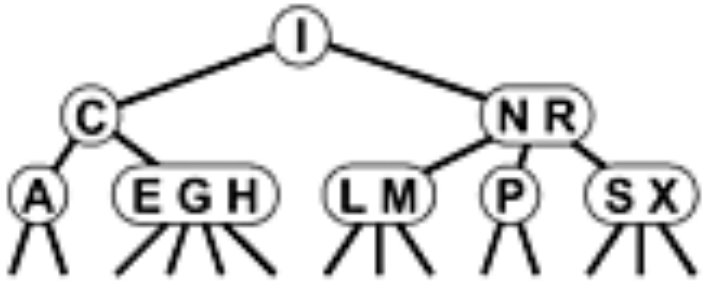
Καθοδικά δέντρα 2-3-4

- Μπορούμε να εγγυηθούμε ότι όλες οι λειτουργίες εισαγωγής και αναζήτησης θα γίνονται πάντα σε χρόνο logN?
- Για να γίνει αυτό χρειαζόμαστε μεγαλύτερη ευελιξία στις δομές που χρησιμοποιούμε
- Τα ***δέντρα 2-3-4*** έχουν τρία είδη κόμβων
  - 2-κόμβοι: 1 κλειδί και 2 σύνδεσμοι (όπως στα ΔΔΑ)
  - 3-κόμβοι: 2 κλειδιά και 3 σύνδεσμοι, π.χ. αν έχει κλειδιά τα 15, 30
    - Πρώτος σύνδεσμος σε υποδέντρο με κλειδιά < 15
    - Δεύτερος σύνδεσμος σε υποδέντρο με κλειδιά στο [15, 30]
    - Τρίτος κόμβος σε υποδέντρο με κλειδιά > 30
  - 4-κόμβοι: 3 κλειδιά και 4 σύνδεσμοι (σε αναλογία με τους 3-κόμβους)

Καθοδικά δέντρα 2-3-4

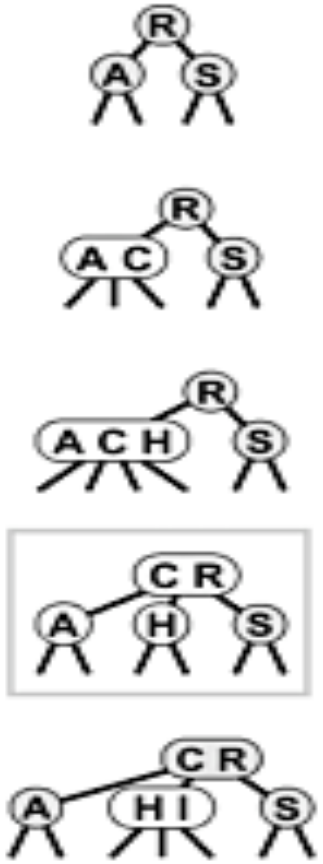
- Ισορροπημένο δένδρο 2-3-4
  - Όλοι οι null σύνδεσμοι ισαπέχουν από τη ρίζα

- Η αναζήτηση είναι γενίκευση αυτής των ΔΔΑ
- Η εισαγωγή είναι λίγο πιο περίπλοκη
  - Για 2-κόμβους και 3-κόμβους αρκεί αναζήτηση και μετά εισαγωγή όπως στα ΔΔΑ
    - 2-κόμβοι μετατρέπονται σε 3-κόμβους και 3-κόμβοι μετατρέπονται σε 4-κόμβους
  - Για 4-κόμβους δεν αρκεί (θέλουμε να διατηρήσουμε την ισορροπία)
  - Τι κάνουμε όταν θέλουμε εισαγωγή σε 4-κόμβο;



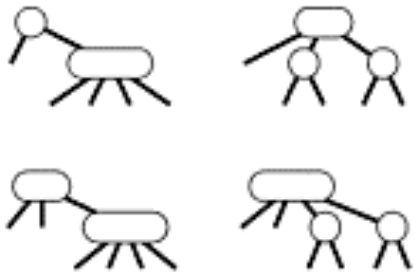
Καθοδικά δέντρα 2-3-4

- Ανοδική εισαγωγή
  - Εντοπίζουμε τον κόμβο στο τελευταίο επίπεδο
  - Αν είναι 4-κόμβος τον σπάμε στα δύο
  - Το μεσαίο κλειδί πηγαίνει στον πατέρα
  - Το νέο κλειδί εισάγεται στο κατάλληλο παιδί
  - Συνεχίζουμε αναδρομικά στον πατέρα



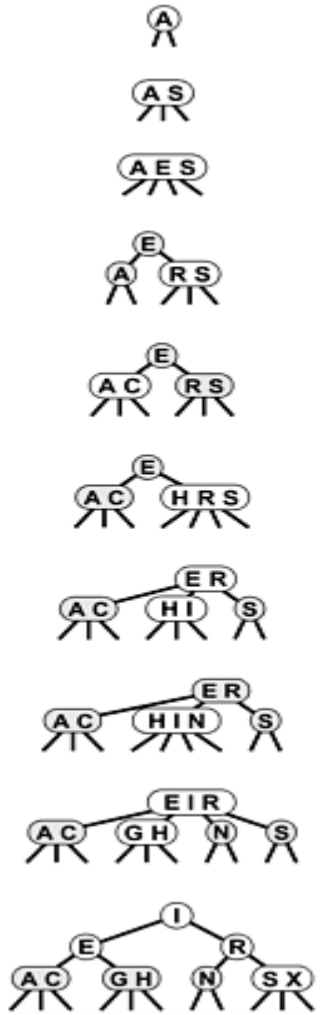
Καθοδικά δέντρα 2-3-4

- Μπορεί όμως να είναι και ο γονέας 4-κόμβος, καθώς και ο γονέας του γονέα, κ.ο.κ.
- Η ανοδική εισαγωγή τότε θα είναι πολύ αργή (μπορεί να αλλάξει όλο το δέντρο)
- Καθοδική εισαγωγή
  - Όποτε βρίσκουμε έναν 4-κόμβο τον σπάμε
    - Αυτό γίνεται σε όλα τα επίπεδα κατεβαίνοντας
  - Εξασφαλίζουμε ότι ένας 4-κόμβος δεν είναι παιδί 4-κόμβου
  - Εισάγουμε το μεσαίο κλειδί στον πατέρα
    - Αν η ρίζα γίνει 4-κόμβος τη διασπάμε άμεσα
    - Δεν περιμένουμε την επόμενη εισαγωγή
  - Εισάγουμε το νέο κλειδί στο τελευταίο επίπεδο



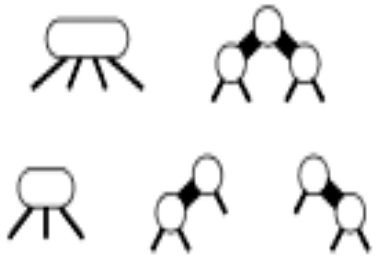
Καθοδικά δέντρα 2-3-4

- Απόδοση καθοδικών δέντρων 2-3-4
  - Αναζήτηση: το πολύ logN+1 διασχίσεις κόμβων
    - Όλοι οι κόμβοι ισαπέχουν από τη ρίζα
    - Μόνο η διάσπαση της ρίζας αυξάνει το ύψος
    - Όλοι οι κόμβοι είναι τουλάχιστον 2-κόμβοι
  - Εισαγωγή: το πολύ logN+1 διαιρέσεις κόμβων
    - Διαιρέσεις σε όλη τη διαδρομή
  - Τα ανοδικά δέντρα θέλουν περισσότερες διαιρέσεις
    - Είναι όμως πιο απλά στην υλοποίηση
- Υλοποίηση δέντρων 2-3-4
  - Η χρήση 3 ειδών κόμβων έχει αρκετό κόστος
  - Μπορεί η επιβάρυνση να υπερβαίνει την ωφέλεια
  - Εναλλακτική υλοποίηση: δέντρα κόκκινου-μαύρου



Δέντρα κόκκινου-μαύρου

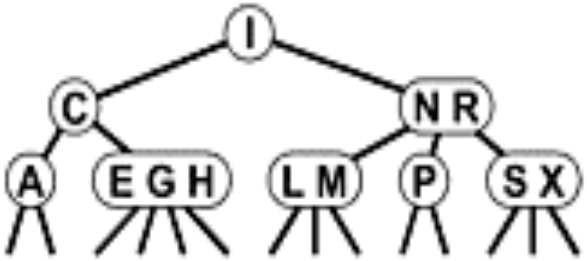
- Δέντρα Κόκκινου-Μαύρου (ΔΚΜ)
  - Αναπαράσταση δέντρων 2-3-4 ως ΔΔΑ
  - Οι 3- και 4-κόμβοι μετατρέπονται σε 2-κόμβους
    - Οι μεταξύ τους σύνδεσμοι είναι κόκκινοι
    - Οι υπόλοιποι σύνδεσμοι είναι μαύροι

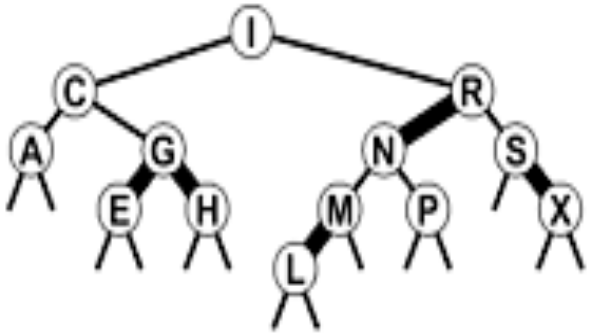


  - Παράσταση 4-κόμβων
    - Τρεις 2-κόμβοι που συνδέονται με κόκκινους συνδέσμους
    - Μεσαίο κλειδί στον πάνω κόμβο
  - Παράσταση 3-κόμβων
    - Δύο 2-κόμβοι που συνδέονται με κόκκινους συνδέσμους
    - Ο κόκκινος σύνδεσμος μπορεί να είναι δεξιά ή αριστερά
  - Ένα bit ανά κόμβο για το χρώμα του συνδέσμου
    - Ορίζει το χρώμα του συνδέσμου με τον πατέρα του

Δέντρα κόκκινου-μαύρου

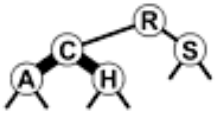
- Αναζήτηση: ακριβώς ίδια με αυτή των ΔΔΑ
- Εισαγωγή: βασίζεται σε αυτή των δέντρων 2-3-4
- Τα δέντρα κόκκινου-μαύρου συνδυάζουν την απλή μέθοδο αναζήτησης των ΔΔΑ με την απλή μέθοδο εισαγωγής-εξισορρόπησης των δέντρων 2-3-4
- Χρειάζονται όμως ένα επιπλέον πεδίο για το χρώμα

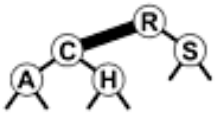


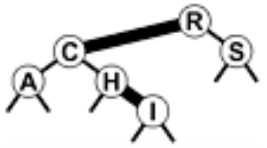


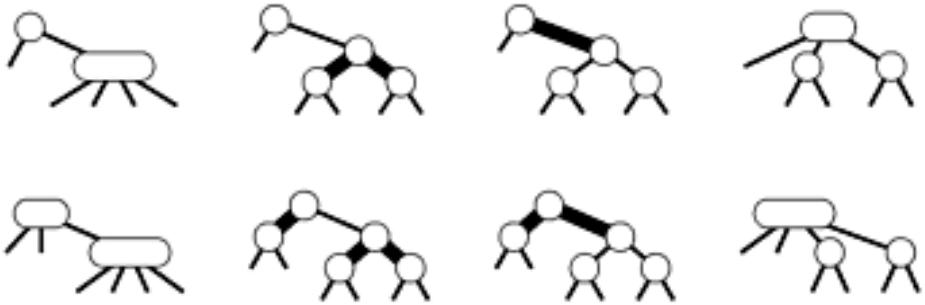
Δέντρα κόκκινου-μαύρου

- Εισαγωγή σε ΔΚΜ
  - Χρησιμοποιούμε τον αλγόριθμο για τα δέντρα 2-3-4
  - Μετατρέπουμε τα είδη κόμβων σε χρώματα συνδέσμων
- Περιπτώσεις διάσπασης 4-κόμβων
  - Ο πατέρας είναι 2-κόμβος
    - Απλή αλλαγή χρωμάτων για να γίνει 3-κόμβος
    - Γίνεται κόκκινος ο σύνδεσμος από τον 2-κόμβο προς το παιδί
  - Ο πατέρας είναι 3-κόμβος με «σωστό» προσανατολισμό
    - Ο κόκκινος σύνδεσμος του 3-κόμβου είναι από την άλλη μεριά
    - Απλή αλλαγή χρωμάτων για να γίνει 4-κόμβος (όπως και πριν)



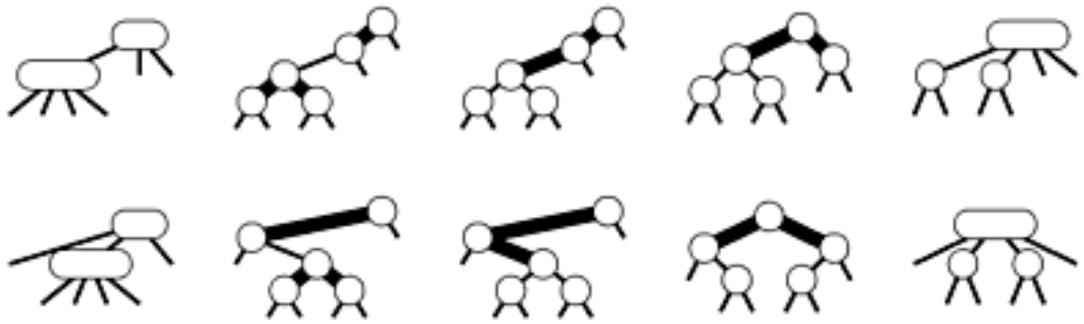






Δέντρα κόκκινου-μαύρου

- Περιπτώσεις διάσπασης 4-κόμβων
  - Ο πατέρας είναι 3-κόμβος με «λάθος» προσανατολισμό
  - Η διαίρεση αφήνει 2 κόκκινους συνδέσμους στη σειρά
    - Η διαίρεση γίνεται με απλή αλλαγή χρωμάτων
    - Το δένδρο δεν αντιστοιχεί πια σε δένδρο 2-3-4
  - Περιστροφή για αποκατάσταση της μορφής
    - Ίδια κατεύθυνση συνδέσμων: απλή περιστροφή
    - Διαφορετική κατεύθυνση συνδέσμων: διπλή περιστροφή



Δέντρα κόκκινου-μαύρου

- Υλοποίηση εισαγωγής σε ΔΚΜ
  - Στην κάθοδο αλλάζουμε τα χρώματα
    - Διασπούμε τους 4-κόμβους
  - Στην άνοδο εκτελούμε τις περιστροφές
    - Το sw δείχνει αν ο τρέχον κόμβος είναι δεξί παιδί
    - Ο σύνδεσμος προς τη ρίζα θεωρείται μαύρος και αριστερός
    - Οι νέοι κόμβοι έχουν κόκκινο σύνδεσμο

```
private static final boolean R = true;
private static final boolean B = false;
private boolean red(Node x) {
  if (x == null) return false;
  return x.cbit; }
void insert(ITEM x) {
  head = insertR(head, x, B); head.cbit = B; }
```

Δέντρα κόκκινου-μαύρου

- Υλοποίηση εισαγωγής σε ΔΚΜ
  - Περίπτωση εισαγωγής στο αριστερό υποδένδρο
    - Διαδοχικοί ανάποδοι κόκκινοι σύνδεσμοι: περιστροφή
    - Διαδοχικοί ίδιοι κόκκινοι σύνδεσμοι: (ξανα)κάνουμε περιστροφή

```
private Node insertR(Node h, ITEM x, boolean sw) {
  if (h == null) { return new Node(x, R); }
  if (red(h.l) && red(h.r)) {
     h.cbit = R; h.l.cbit = B; h.r.cbit = B; }
  if (less(x.key(), h.item.key())) {
     h.l = insertR(h.l, x, false);
     if (red(h) && red(h.l) && sw) h = rotR(h);
     if (red(h.l) && red(h.l.l)) {
        h = rotR(h); h.cbit = B; h.r.cbit = R; } }
  else { …
```

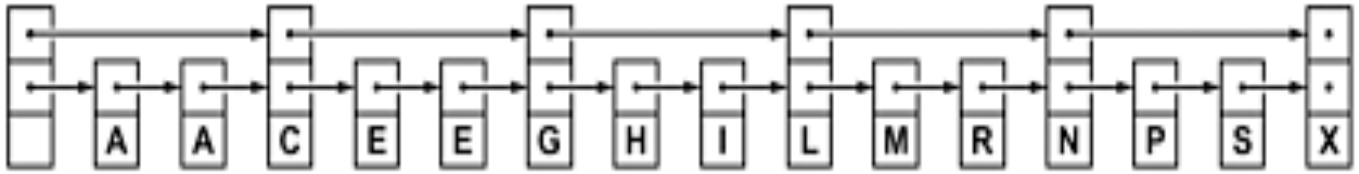
Δέντρα κόκκινου-μαύρου

- Υλοποίηση εισαγωγής σε ΔΚΜ
  - Περίπτωση εισαγωγής στο δεξί υποδένδρο
    - Συμμετρική με την προηγούμενη
    - Η πρώτη περιστροφή ανάγει τη μία περίπτωση στην άλλη

```
… else {
      h.r = insertR(h.r, x, true);
      if (red(h) && red(h.r) && !sw) h = rotL(h);
      if (red(h.r) && red(h.r.r)) {
         h = rotL(h); h.cbit = B; h.l.cbit = R; } }
  return h; }
```

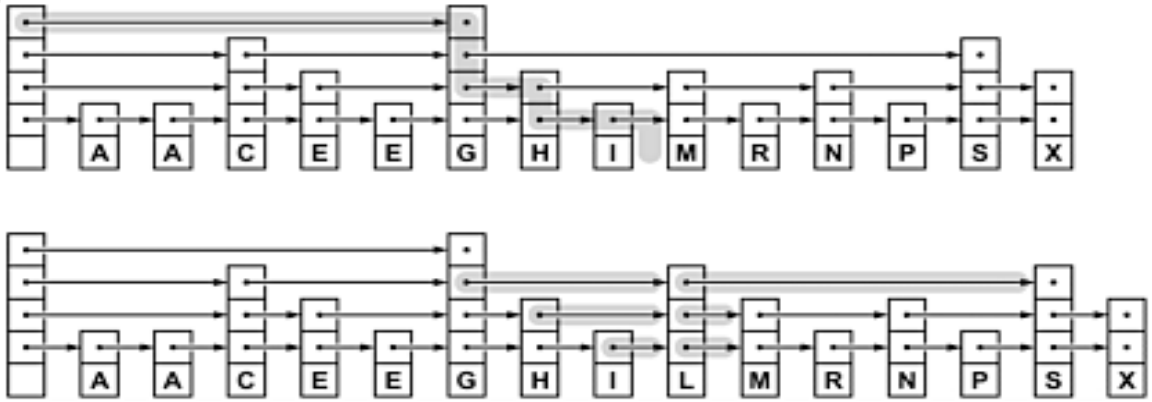
- Απόδοση αναζήτησης σε ΔΚΜ
  - Χειρότερη περίπτωση: το πολύ 2logN+2 συγκρίσεις
  - Μέση περίπτωση: 1,002logN συγκρίσεις
    - Υποθέτουμε τυχαία κλειδιά κατά την κατασκευή

Λίστες παράλειψης (skip lists)



- Ταξινομημένη λίστα
- Χρήση πρόσθετων συνδέσμων για να παραλείπουμε μεγάλα τμήματα της λίστας κατά την αναζήτηση
- Κάθε κόμβος μπορεί να έχει μεταβλητό πλήθος συνδέσμων
- Κάθε σύνδεσμος μπορεί να παραλείπει μεταβλητό πλήθος κομβων

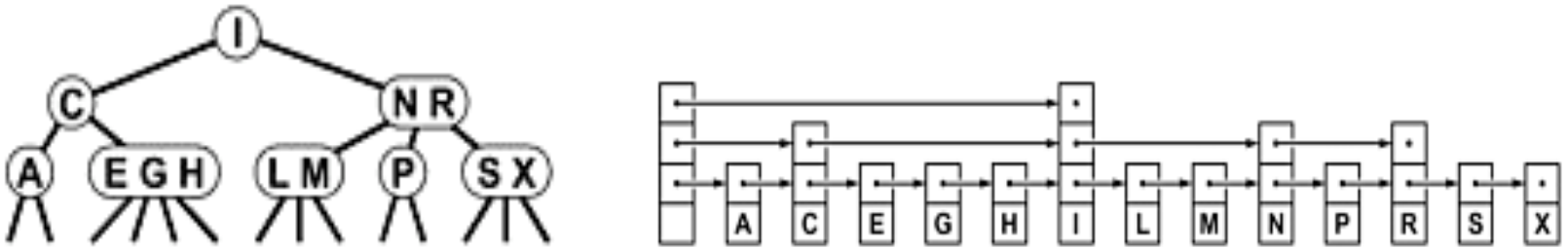
Λίστες παράλειψης (skip lists)



- Αναζήτηση: ξεκινάμε από το πιο πάνω επίπεδο μέχρι να βρούμε σύνδεσμο που δείχνει σε κόμβο με μεγαλύτερο κλειδί από αυτό που ψάχνουμε
- Πηγαίνουμε στο επόμενο επίπεδο συνδέσμων και επαναλαμβάνουμε τη διαδικασία
- Διατήρηση πίνακα συνδέσμων σε κάθε κόμβο (μπορούμε και τυχαιοποιημένα να αποφασίσουμε τον αριθμό των συνδέσμων σε κάθε κόμβο χωριστά)

Λίστες παράλειψης (skip lists)

- Οι λίστες παράλειψης είναι μία εναλλακτική αναπαράσταση των ισορροποημένων δέντρων
- Οι αλγόριθμοι για δέντρα 2-3-4 μπορούν να υλοποιηθούν με λίστες παράλειψης αντί για δέντρα κόκκινου-μαύρου
- Αντιστοίχιση: για κάθε κόμβο πλήθος συνδέσμων = ύψος του κόμβου στο δέντρο
- Σύνδεση κόμβων οριζόντια (ανά επίπεδο)
- Λίγο πιο πολύπλοκος κώδικας



Χαρακτηριστικά Επιδόσεων Ισορροπημένων Δέντρων

- Τυχαιοποιημένα ΔΔΑ: πιο εύκολη υλοποίηση, πρέπει να έχουμε καλή γεννήτρια τυχαίων αριθμών
- Στρεβλά ΔΔΑ: άμεση επέκταση του αλγορίθμου εισαγωγής στη ρίζα
- Δέντρα κόκκινου-μαύρου: καλύτερες εγγυήσεις για γρήγορη αναζήτηση
- Λίστες παράλειψης: κατάλληλες για πλήρες φάσμα λειτουργιών πίνακα συμβόλων, πιο αργές οι εισαγωγές και αναζητήσεις

Χαρακτηριστικά Επιδόσεων Ισορροπημένων Δέντρων

Πειραματικά δεδομένα – Κατασκευή δέντρου

N	ΔΔΑ	Τυχ. ΔΔΑ	Splay ΔΔΑ	R-B	Skip Lists
12500	90	197	267	92	145
25000	167	492	215	181	385
50000	381	1105	1125	430	892
100000	1004	2656	1148	1190	3257
200000	2628	6341	2784	2936	7493

Χαρακτηριστικά Επιδόσεων Ισορροπημένων Δέντρων

Πειραματικά δεδομένα – Αναζητήσεις

N	ΔΔΑ	Τυχ. ΔΔΑ	Splay ΔΔΑ	R-B	Skip Lists
12500	75	86	80	60	145
25000	175	212	195	158	355
50000	420	505	433	359	878
100000	1047	1357	1113	861	2094
200000	2553	2893	2649	2114	5109