

Διαδικά δένδρα αναζήτησης

Γιώργος Στάμου

Εργαστήριο Συστημάτων Τεχνητής Νοημοσύνης και Μάθησης
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Εθνικό Μετσόβιο Πολυτεχνείο



Εισαγωγή στα δυαδικά δένδρα αναζήτησης

- Μέχρι τώρα θεωρήσαμε δένδρα στα οποία οι **σχετικές θέσεις** των κόμβων μέσα στο δένδρο δεν έχουν περιορισμούς.
- Δηλαδή, ένα αντικείμενο μπορεί να εμφανίζεται οπουδήποτε μέσα στο δένδρο, δε γνωρίζουμε που ακριβώς βρίσκεται το αντικείμενο μέσα στο δένδρο.
- Μπορεί να χρειαστεί να κάνουμε μία **πλήρη** διάσχιση του δένδρου (στη χειρότερη περίπτωση) προκειμένου να το βρούμε.
- Προκειμένου να διευκολύνουμε την αναζήτηση, μπορούμε να βάλουμε **περιορισμούς** ως προς τις σχετικές θέσεις των αντικειμένων μέσα στο δένδρο.

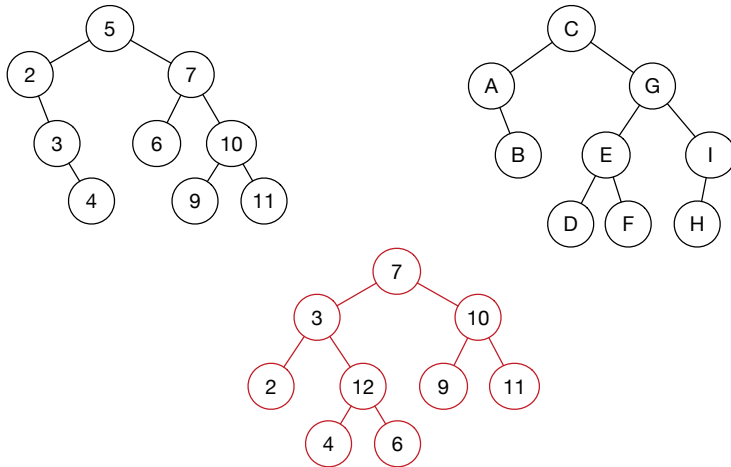
Δένδρα αναζήτησης

- Ένα δένδρο το οποίο υποστηρίζει αποδοτικές λειτουργίες αναζήτησης, εισαγωγής και διαγραφής ονομάζεται **δένδρο αναζήτησης**.
- Χρησιμοποιείται προκειμένου να αποθηκεύσει ένα πεπερασμένο σύνολο κλειδιών που προέρχονται από ένα **ολικά διατεταγμένο** σύνολο κλειδιών K .
- Αυτό που κάνει ένα δένδρο να είναι δένδρο αναζήτησης είναι το ότι τα κλειδιά δεν εμφανίζονται σε αυθαίρετους κόμβους του δένδρου: υπάρχει ένα **κριτήριο διάταξης δεδομένων** το οποίο καθορίζει το πού μπορεί να εμφανιστεί ένα κλειδί μέσα στο δένδρο σε σχέση με τα άλλα κλειδιά του δένδρου.

Δυαδικό δένδρο αναζήτησης – Ορισμός

- Ένα **δυαδικό δένδρο αναζήτησης** T (Binary Search Tree - BST) είναι ένα δυαδικό δένδρο $T = \{r; T_L, T_R\}$ με τις εξής ιδιότητες:
 - Όλοι οι κόμβοι που περιέχονται στο αριστερό υποδένδρο, T_L , έχουν ρίζες μικρότερες από το r , δηλαδή $\forall k \in T_L: k < r$.
 - Όλοι οι κόμβοι που περιέχονται στο δεξί υποδένδρο, T_R , έχουν ρίζες μεγαλύτερες από το r , δηλαδή $\forall k \in T_R: k > r$.
 - Τα T_L, T_R είναι δυαδικά δένδρα αναζήτησης.

Παραδείγματα BST



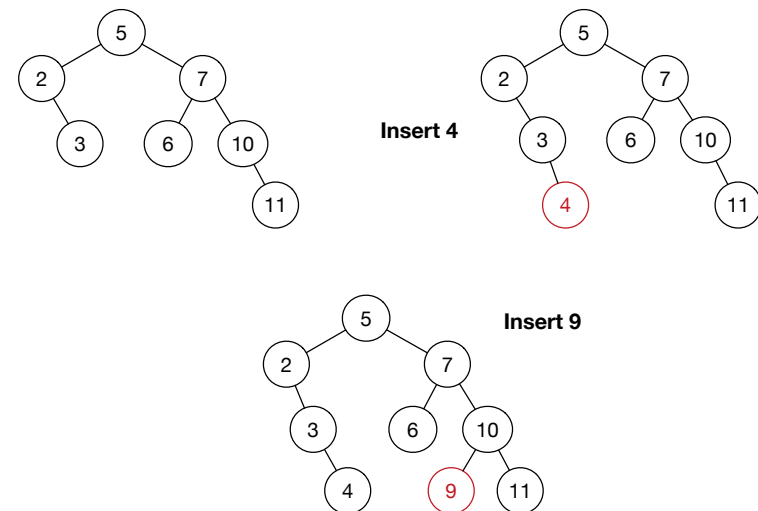
Αναζήτηση στοιχείων σε BST

- Η αναζήτηση ξεκινά από τη **ρίζα** του δένδρου.
- Εάν το στοιχείο που αναζητούμε, x , είναι ίσο με τη ρίζα r , η αναζήτηση τερματίζεται επιτυχώς.
- Εάν όχι, τότε, εάν το x είναι **μικρότερο** από το r , θα γίνει αναζήτηση στο **αριστερό υποδένδρο**, αλλιώς, το x πρέπει να είναι **μεγαλύτερο** από το r και στην περίπτωση αυτή θα γίνει αναζήτηση στο **δεξί υποδένδρο**.

Εισαγωγή στοιχείων σε BST

- Ακολουθούμε το μονοπάτι που θα ακολουθούσε η μέθοδος αναζήτησης, προκειμένου να προσδιορίσουμε πού θα ήταν το στοιχείο (εάν αυτό υπήρχε στο δένδρο).
- Υποθέτοντας ότι το στοιχείο δεν υπάρχει ήδη στο δένδρο, η αναζήτηση θα είναι **ανεπιτυχής** και θα τερματίσει σε ένα **κενό υποδένδρο**.
- Αυτή ακριβώς είναι η **θέση** στην οποία θα πρέπει να εισαχθεί το νέο στοιχείο (ως φύλλο αντί για το κενό υποδένδρο)!

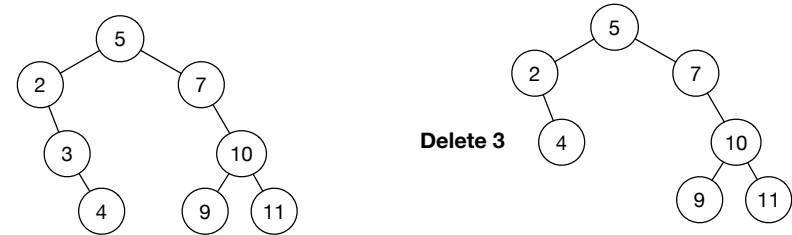
Εισαγωγή στοιχείων σε BST



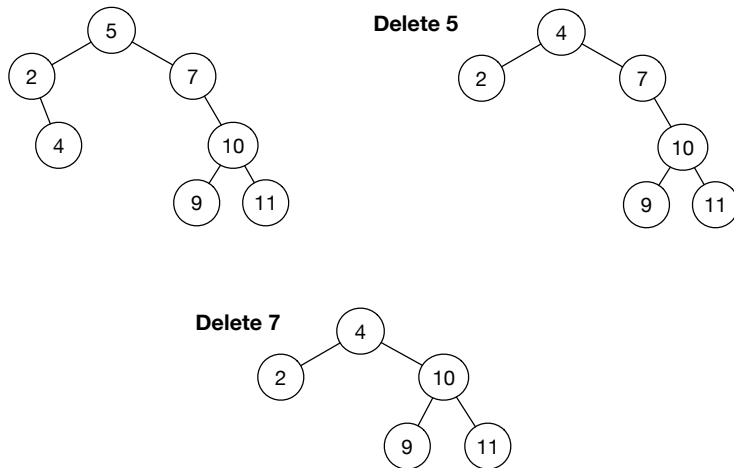
Διαγραφή στοιχείων από BST

- Ακολουθούμε το μονοπάτι αναζήτησης, προκειμένου να προσδιορίσουμε το στοιχείο (εάν υπάρχει στο δένδρο).
- Αν το στοιχείο υπάρχει στο δένδρο, η αναζήτηση θα είναι **επιτυχής** τερματίζοντας στον **κόμβο** που θα πρέπει να διαγραφεί.
- Αν ο κόμβος είναι **φύλλο** τον διαγράφουμε, μετατρέποντάς τον σε κενό υποδένδρο.
- Αν ο κόμβος αυτός έχει **ένα παιδί** τον διαγράφουμε, αντικαθιστώντας τον με το μη κενό παιδί του.
- Αν έχει **δύο παιδιά**, αντικαθιστούμε τη ρίζα του με τη ρίζα του προηγούμενου κόμβου στην ενδοδιατεταγμένη διάσχιση και διαγράφουμε εκείνο τον κόμβο (ο οποίος έχει ένα παιδί ή είναι φύλλο)

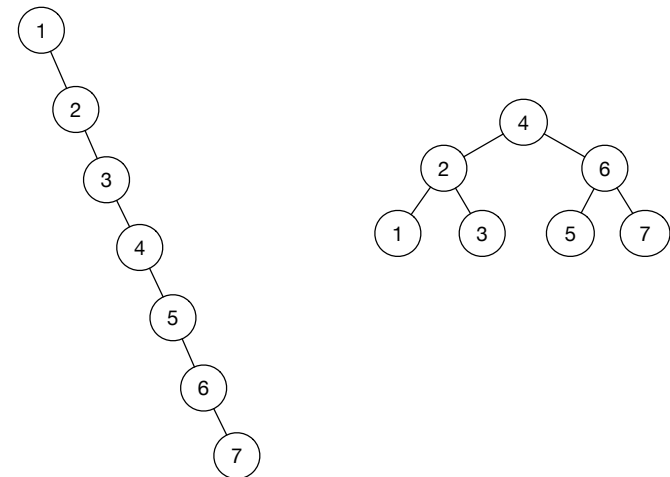
Διαγραφή στοιχείων από BST



Διαγραφή στοιχείων από BST



Παράδειγμα αναζήτησης σε BST – Πολυπλοκότητα



Τέλειο δυαδικό δένδρο – Ορισμός

- Ένα τέλειο δυαδικό δένδρο ύψους $h \geq 0$ είναι ή το κενό ή ένα δυαδικό δένδρο $T = \{r; T_L, T_R\}$ με τις παρακάτω ιδιότητες:
 - Εάν $h=0$, τότε T_L και T_R κενά.
 - Αλλιώς, αν $h>0$, τα T_L και T_R είναι και τα δύο τέλεια δυαδικά δένδρα ύψους $h - 1$.

Τέλειο δυαδικό δένδρο – Ιδιότητες

- Είναι σχετικά εύκολο να δείξουμε ότι ένα τέλειο δυαδικό δένδρο ύψους h έχει ακριβώς $2^{h+1} - 1$ κόμβους.
- Αντιστρόφως, το ύψος ενός τέλειου δυαδικού δένδρου με n εσωτερικούς κόμβους είναι $\log_2(n + 1) - 1$.
- Εάν έχουμε ένα δένδρο αναζήτησης το οποίο έχει το σχήμα ενός τέλειου δυαδικού δένδρου, τότε κάθε ανεπιτυχής αναζήτηση επισκέπτεται ακριβώς $h+1$ το πλήθος εσωτερικούς κόμβους.
- Ο χρόνος εκτέλεσης για μία ανεπιτυχή αναζήτηση σε ένα τέλειο δένδρο, στην χειρότερη περίπτωση είναι της τάξης $O(\log n)$.

Τέλειο δυαδικό δένδρο – Ιδιότητες

- Συνεπώς η συνθήκη του τέλειου δένδρου διασφαλίζει αποτελεσματικές και αποδοτικές αναζητήσεις.
- Μπορούμε να κατασκευάσουμε τέλεια δένδρα με n κόμβους **μόνο** για $n = 1, 3, 7, 15, 31, 63, \dots$
- Άρα, τελικά η συνθήκη του τέλειου δένδρου είναι μία ακατάλληλη (πρακτικά) συνθήκη ισορροπίας, επειδή δεν μπορούμε να κατασκευάσουμε ένα ισορροπημένο δένδρο για κάθε τιμή του n .

Συνθήκες ισορροπίας για BST

- Ποια είναι τα χαρακτηριστικά μίας καλής *συνθήκης ισορροπίας*;
- Μία καλή συνθήκη ισορροπίας εξασφαλίζει ότι το ύψος ενός δένδρου με n κόμβους είναι της τάξης $O(\log n)$.
- Η καλή συνθήκη ισορροπίας πρέπει να μπορεί να διατηρηθεί αποδοτικά.
- Δηλαδή η πρόσθετη δουλειά η οποία είναι απαραίτητη προκειμένου να ισορροπήσει το δένδρο, όταν ένα στοιχείο εισάγεται ή διαγράφεται, πρέπει να είναι της τάξης $O(\log n)$.