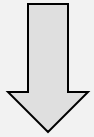
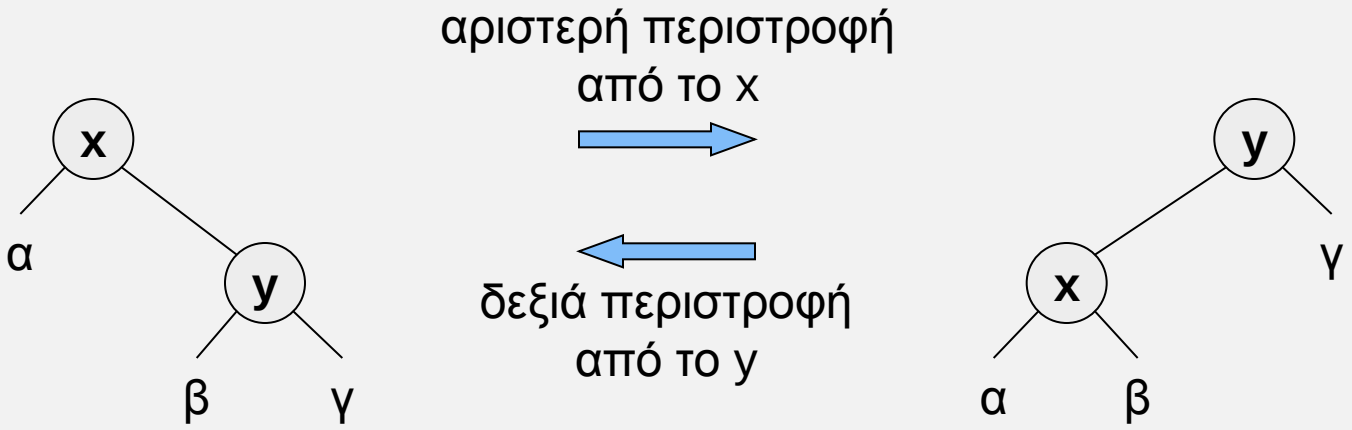
Ισορροπημένα Δένδρα

Μπορούμε να επιτύχουμε $O(\log N)$ χρόνο εκτέλεσης για κάθε λειτουργία;



Ισορροπημένο δένδρο : Διατηρεί ύψος $O(\log N)$ μετά από κάθε εισαγωγή ή διαγραφή

Περιστροφές




```
Node rotateLeft(Node x)
{
     Node y = x.right;
     x.right = y.left;
     y.left = x;
     return y;
}
```

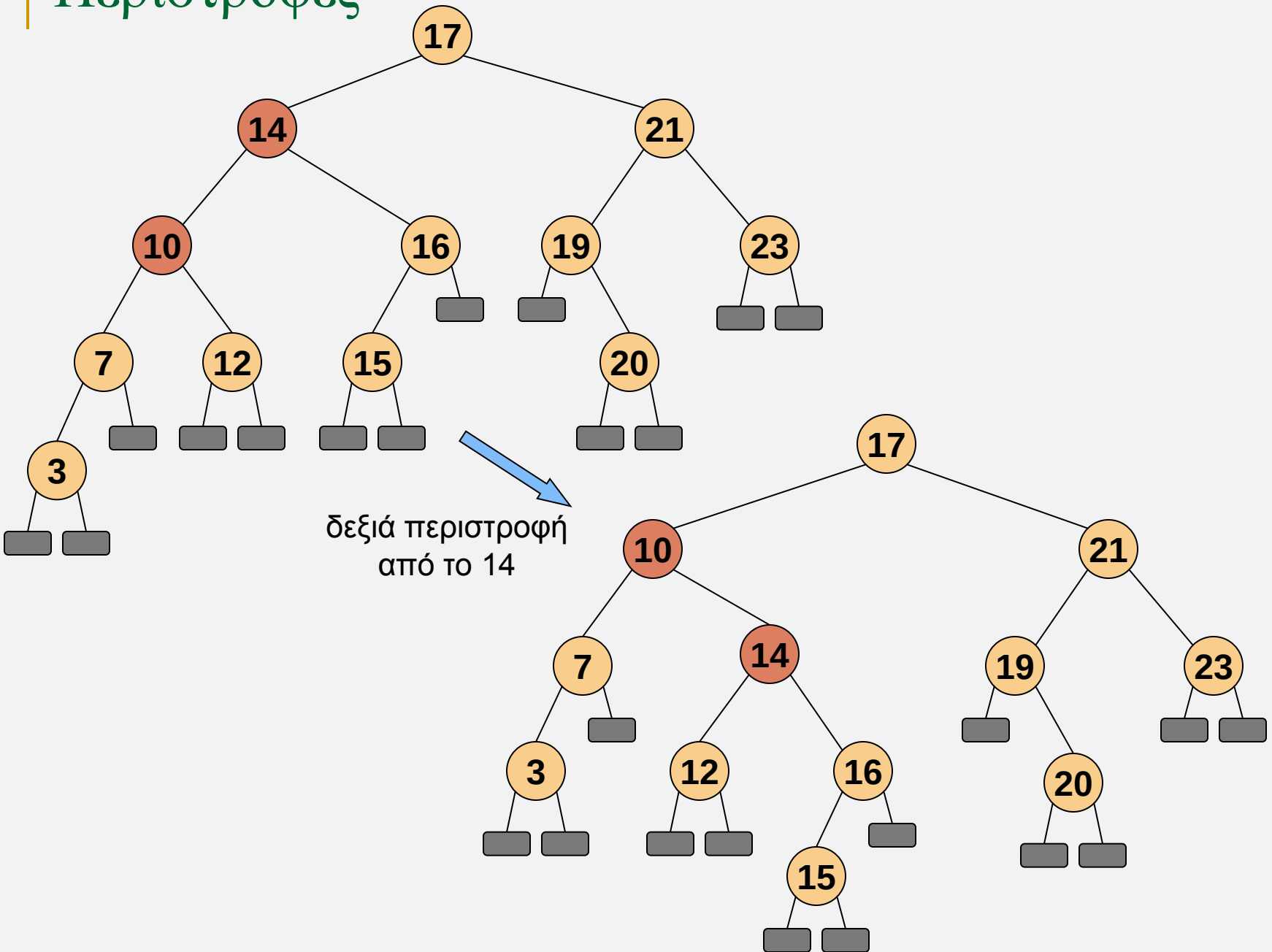
αριστερή περιστροφή

Η περιστροφή παίρνει χρόνο O(1)

```
Node rotateRight(Node y)
{
     Node x = y.left;
     y.left = x.right;
     x.right = y;
     return x;
}
```

δεξιά περιστροφή

Περιστροφές



Ισορροπημένα Δένδρα

Μερικοί τύποι ισορροπημένων δένδρων

- Τυχαιοποιημένα δένδρα (*)
- Αρθρωτά δένδρα (splay trees)
- Δένδρα AVL
- Δένδρα κόκκινου-μαύρου
- (a,b) δένδρα

Όλα τα παραπάνω χρησιμοποιούν περιστροφές για να παραμείνουν ισορροπημένα

(*) Τα τυχαιοποιημένα δένδρα είναι ισορροπημένα με μεγάλη πιθανότητα

Εισαγωγή στη ρίζα

1. Εκτελούμε τον αλγόριθμο εισαγωγής όπως σε ένα απλό δυαδικό δένδρο.

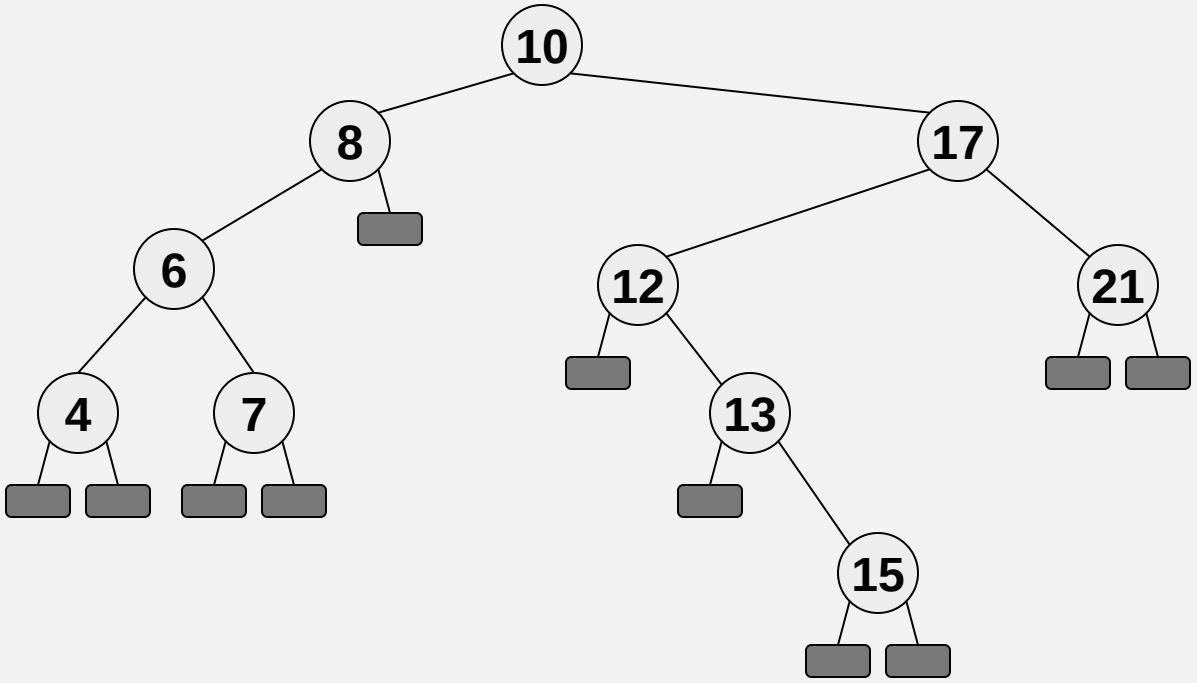
2. Μεταφέρουμε το νέο στοιχείο στη ρίζα του δένδρου με τη χρήση περιστροφών.

```
Node insertIntoRoot(Node x, Key k, Item i)
{
   if (x==null) return new Node(k,i,1);
   int c = k.compareTo(x.key);
   if (c<0) {
       x.left  = insertIntoRoot(x.left,k,i);
       x = rotateRight(x); }
   else if (c>0) {
       x.right = insertIntoRoot(x.right,k,i);
       x = rotateLeft(x); }
   else x.item = i;
   return x;
}

void insertIntoRoot(Key k, Item i) {
   root = insertToRoot(root,k,i);
}
```

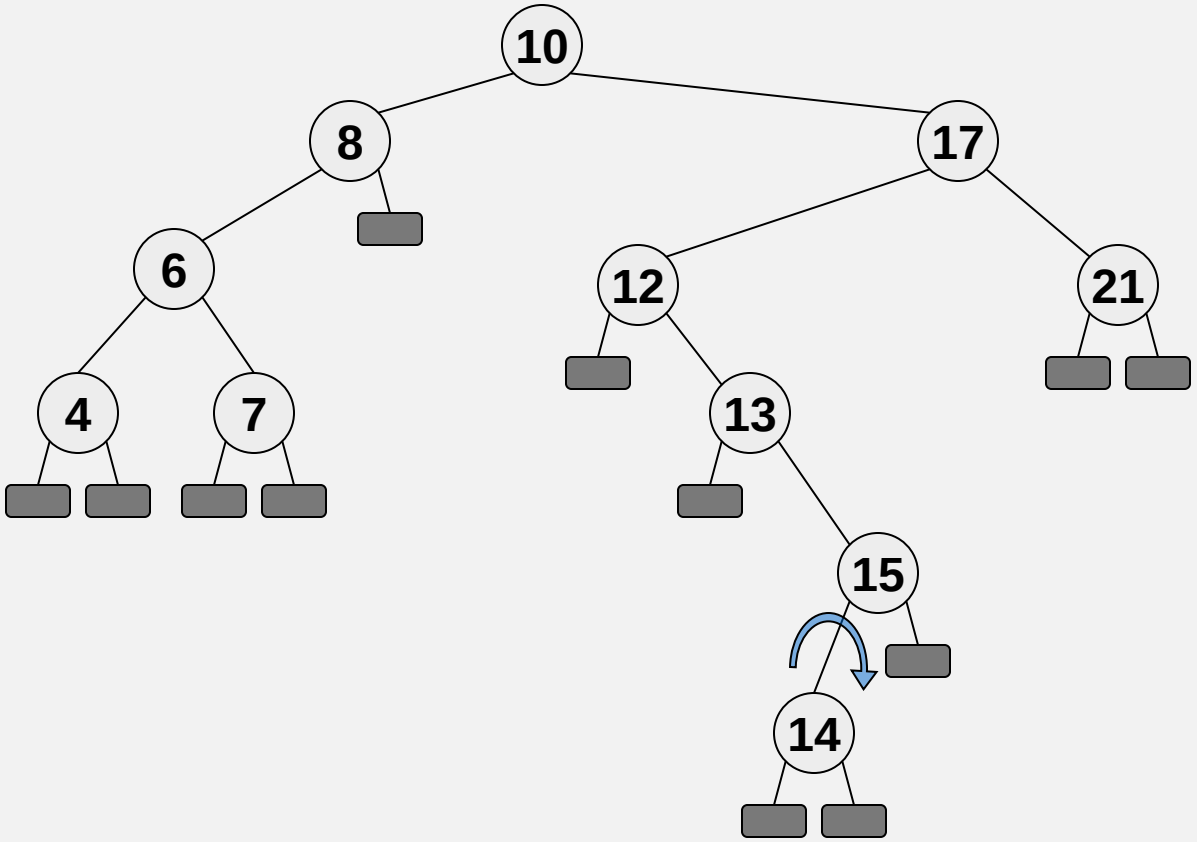
Εισαγωγή στη ρίζα

Εισαγωγή 14



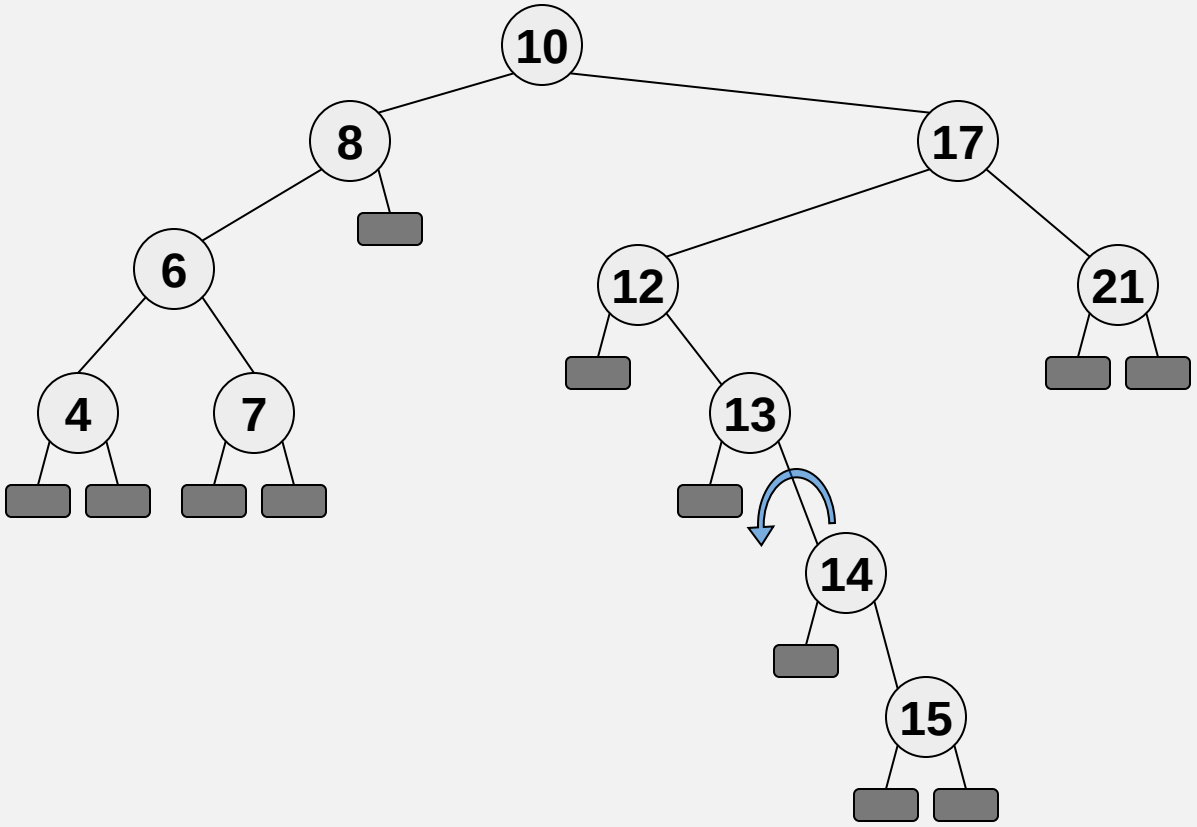
Εισαγωγή στη ρίζα

Περιστροφή 15



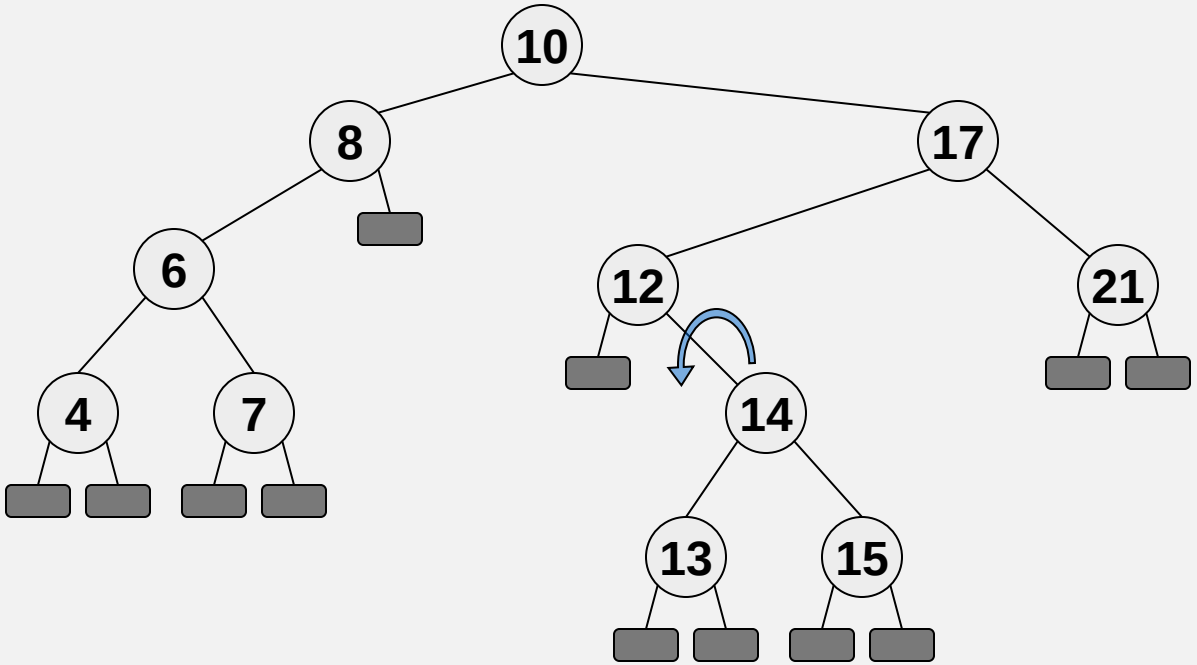
Εισαγωγή στη ρίζα

Περιστροφή 13



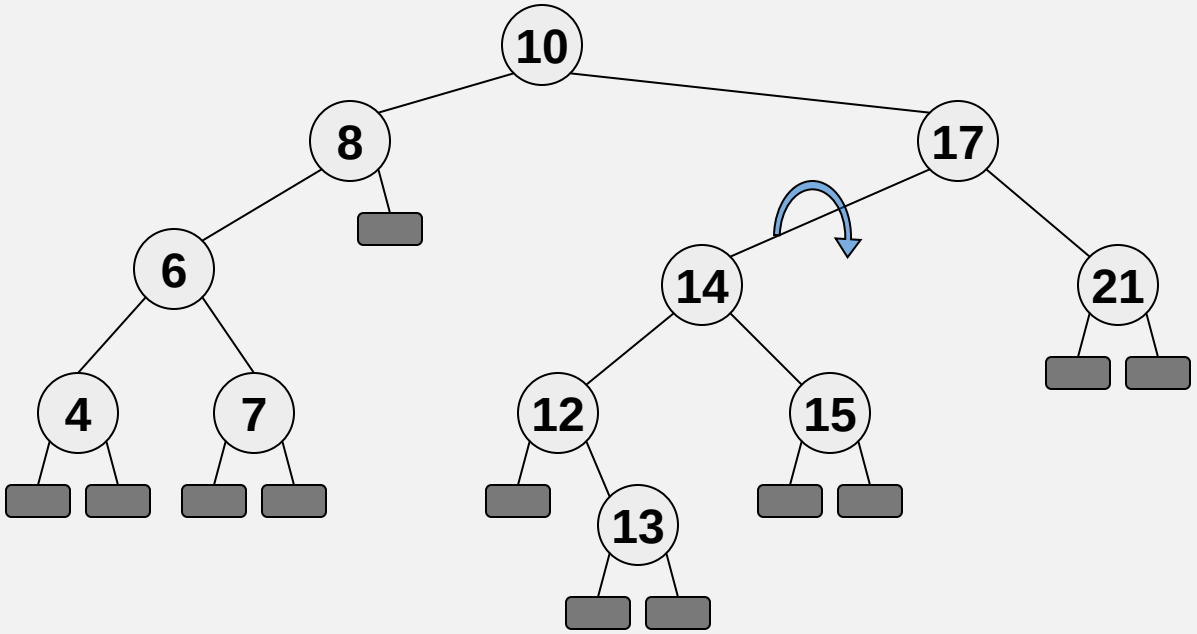
Εισαγωγή στη ρίζα

Περιστροφή 12



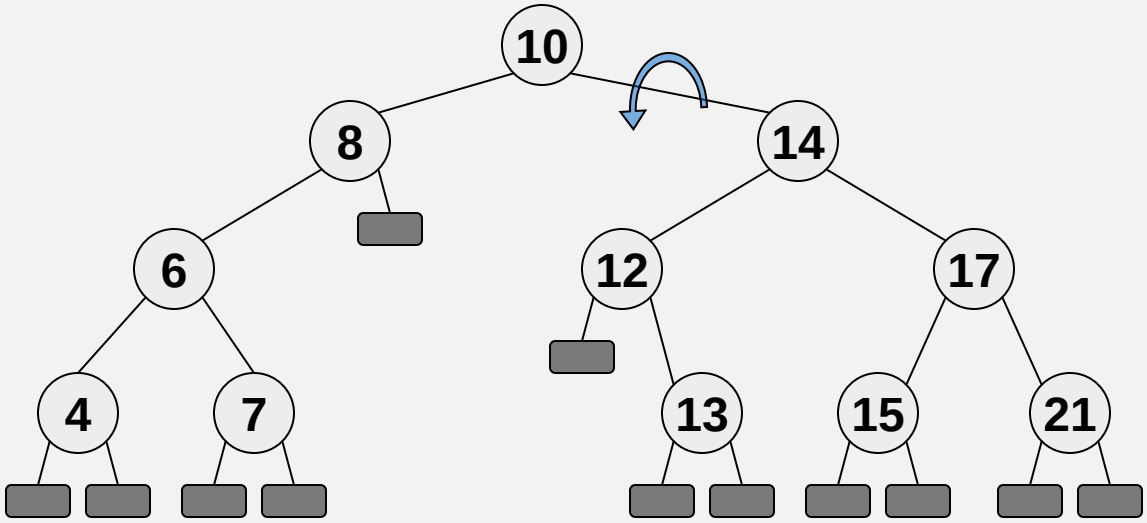
Εισαγωγή στη ρίζα

Περιστροφή 17

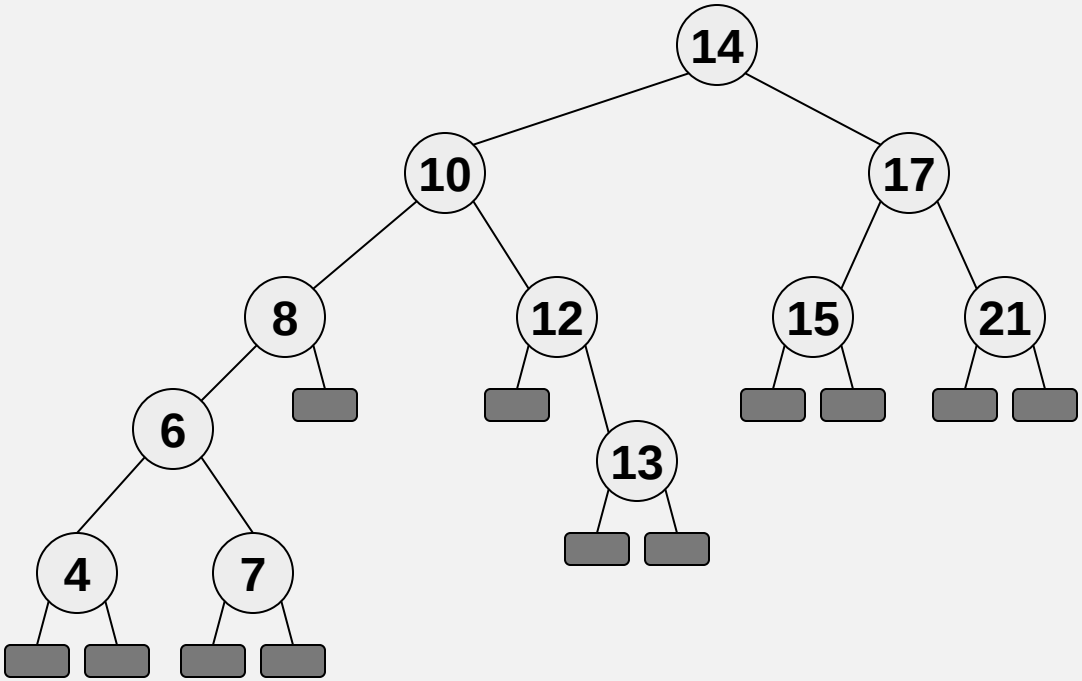


Εισαγωγή στη ρίζα

Περιστροφή 10



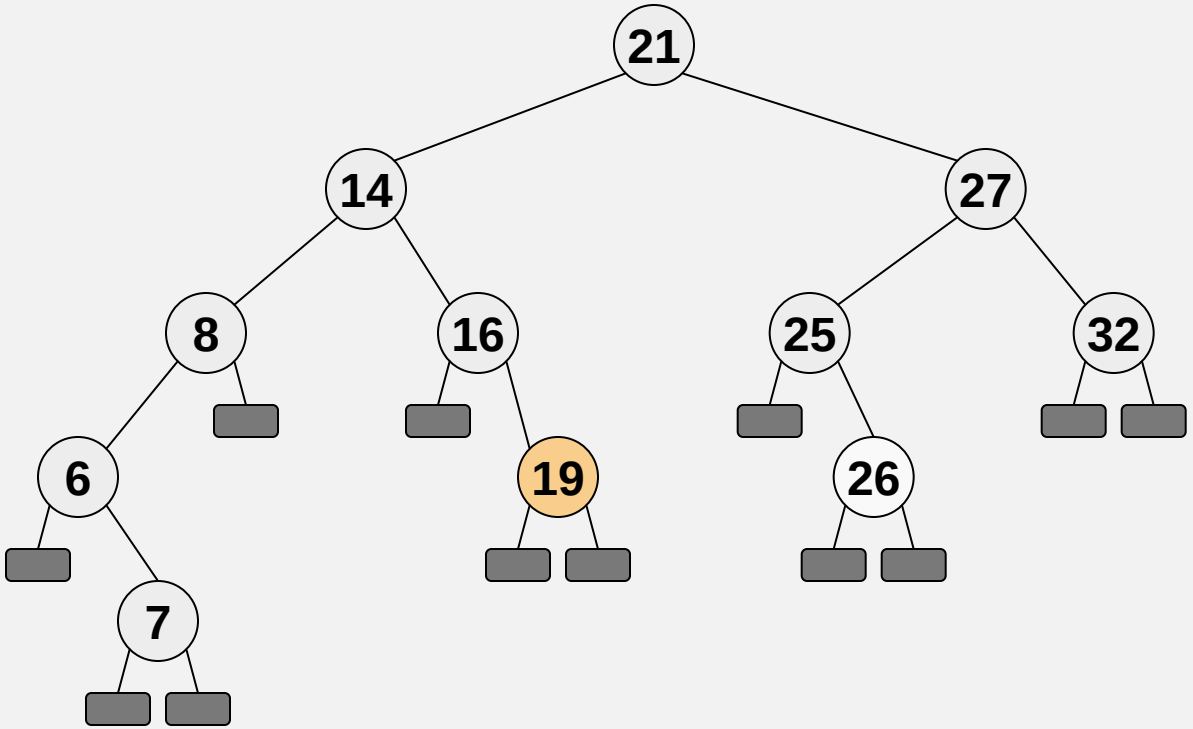
Εισαγωγή στη ρίζα



Διαμέριση

Τοποθετεί τον κόμβο με κλειδί k στη θέση της ρίζας του δένδρου.

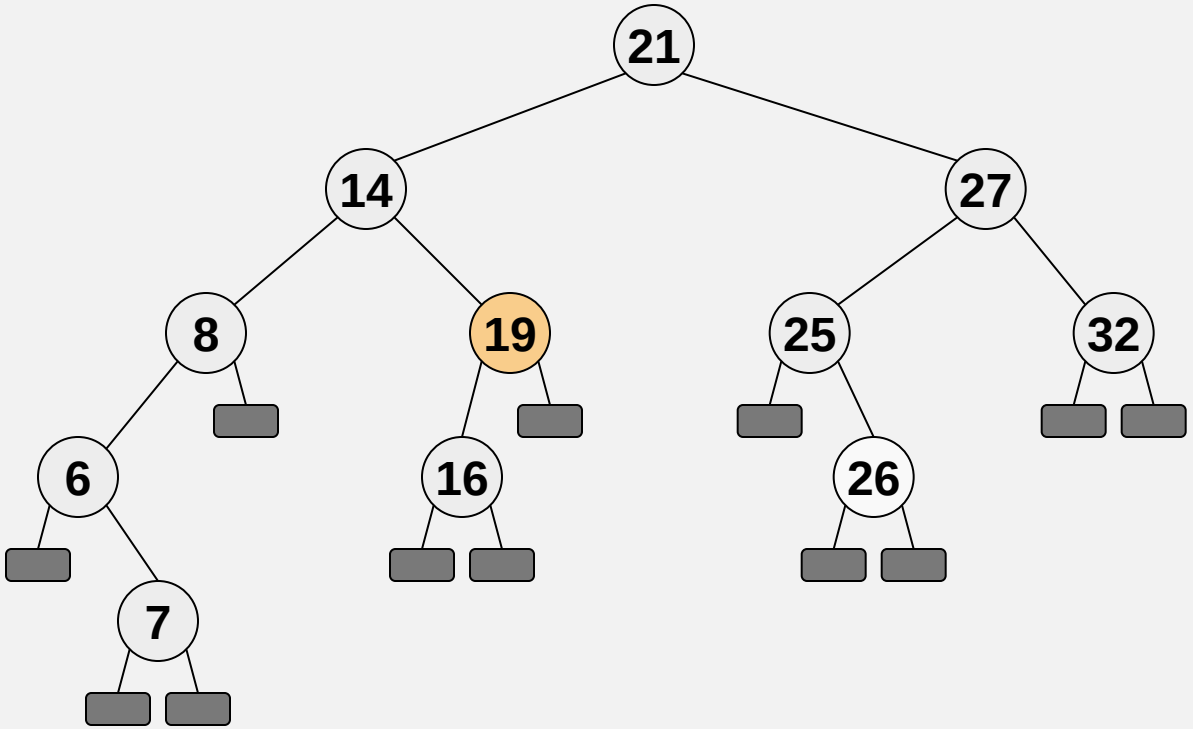
Η μεταφορά του κόμβου γίνεται με τη χρήση περιστροφών.



Διαμέριση

Τοποθετεί τον κόμβο με κλειδί k στη θέση της ρίζας του δένδρου.

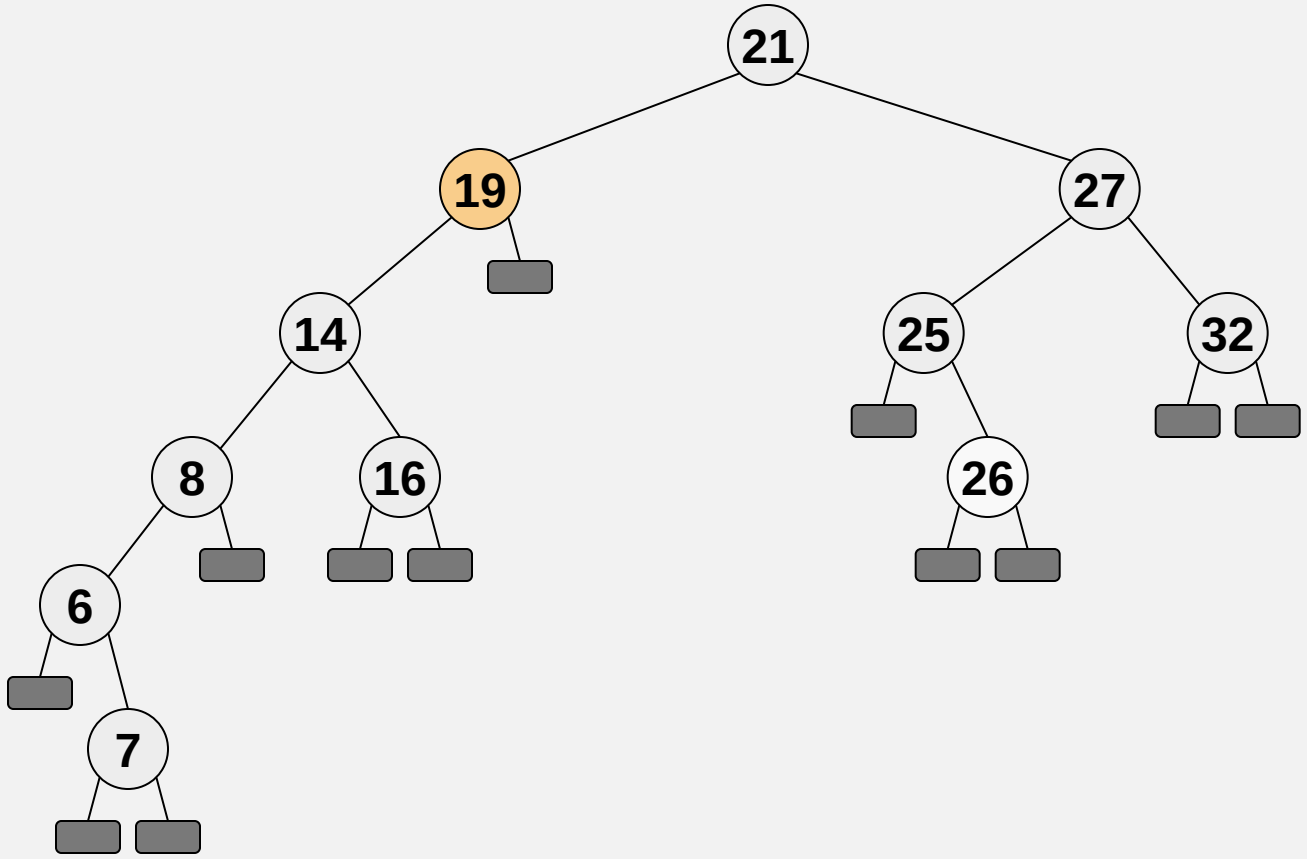
Η μεταφορά του κόμβου γίνεται με τη χρήση περιστροφών.



Διαμέριση

Τοποθετεί τον κόμβο με κλειδί k στη θέση της ρίζας του δένδρου.

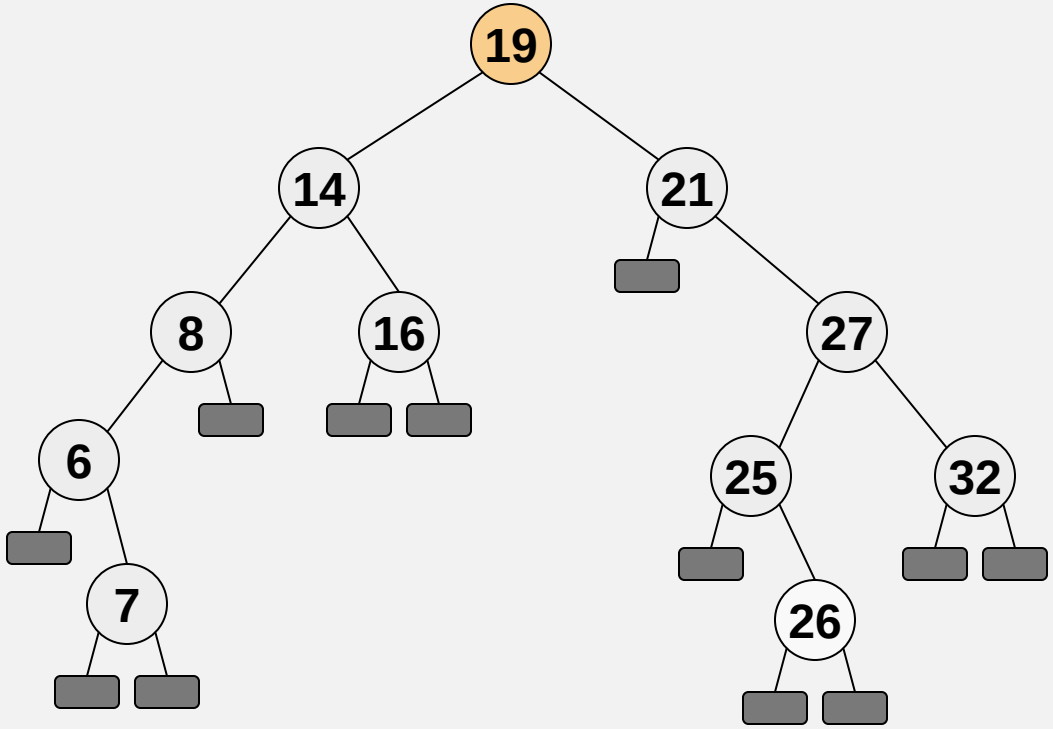
Η μεταφορά του κόμβου γίνεται με τη χρήση περιστροφών.



Διαμέριση

Τοποθετεί τον κόμβο με κλειδί k στη θέση της ρίζας του δένδρου.

Η μεταφορά του κόμβου γίνεται με τη χρήση περιστροφών.



Χωρίζει το σύνολο των κλειδιών σε αυτά που είναι μικρότερα του k και βρίσκονται στο αριστερό υποδένδρο και σε αυτά που είναι μεγαλύτερα του k και βρίσκονται στο δεξί υποδένδρο

Τυχαιοποιημένα δένδρα

Εισαγωγή ενός νέου στοιχείου x σε δένδρο με N στοιχεία

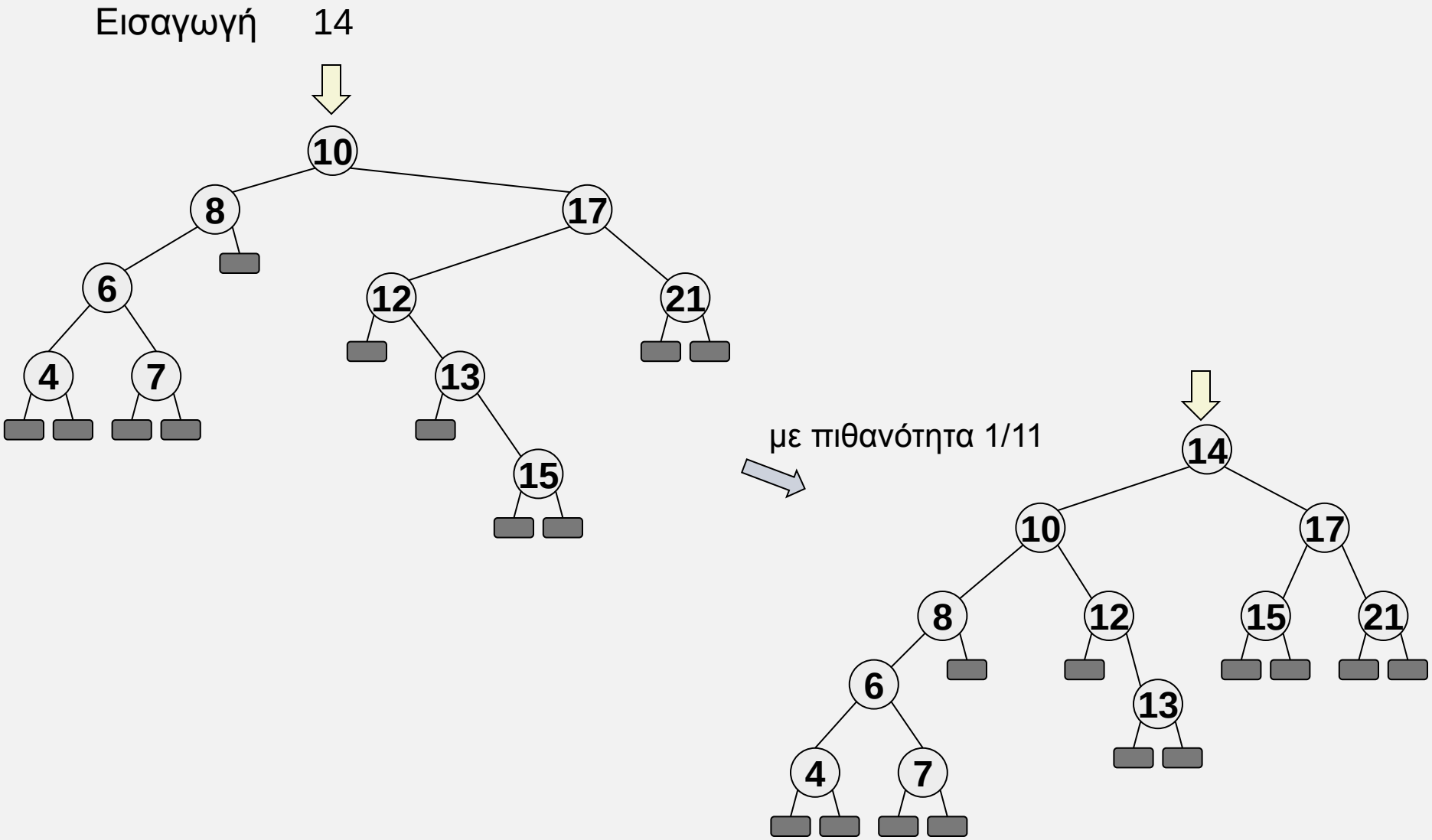
1. Εκτελούμε τον αλγόριθμο εισαγωγής στη ρίζα με πιθανότητα $\frac{1}{N+1}$

2. Διαφορετικά εισάγουμε αναδρομικά το νέο στοιχείο στο κατάλληλο υποδένδρο:
   - Αν το x < κλειδί της ρίζας καλούμε αναδρομικά την εισαγωγή για το αριστερό υποδένδρο
   - Αν το x > κλειδί της ρίζας καλούμε αναδρομικά την εισαγωγή για το δεξιό υποδένδρο

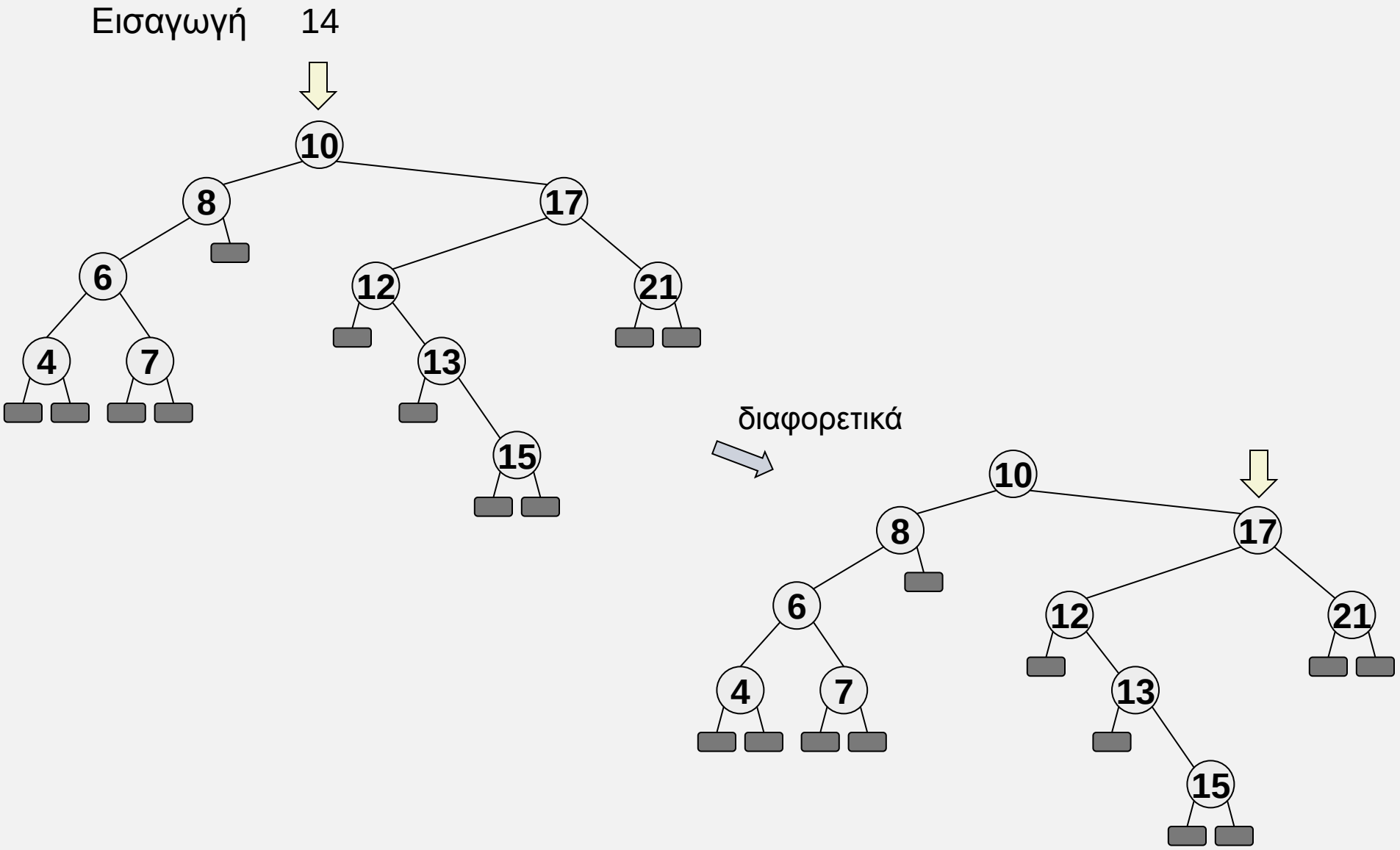
```
Random rand = new Random(seed);
...
Node insert(Node x, Key k, Item i) {
   if (x==null) return new Node(k,i,1);
   if (rand.nextInt(x.N+1) == 0) return insertIntoRoot(x,k,i);
   int c = k.compareTo(x.key);  // σύγκριση με το κλειδί του x
   if      (c < 0) x.left  = insert(x.left, k,i);
   else if (c > 0) x.right = insert(x.right,k,i);
   else x.item = i;  // το αντικείμενο i έχει κλειδί k==x.key
   x.N = size(x.left) + size(x.right) + 1; return x;
}

void insert(Key k, Item i) { root = insert(root,k,i); }
```

Εισαγωγή στη ρίζα

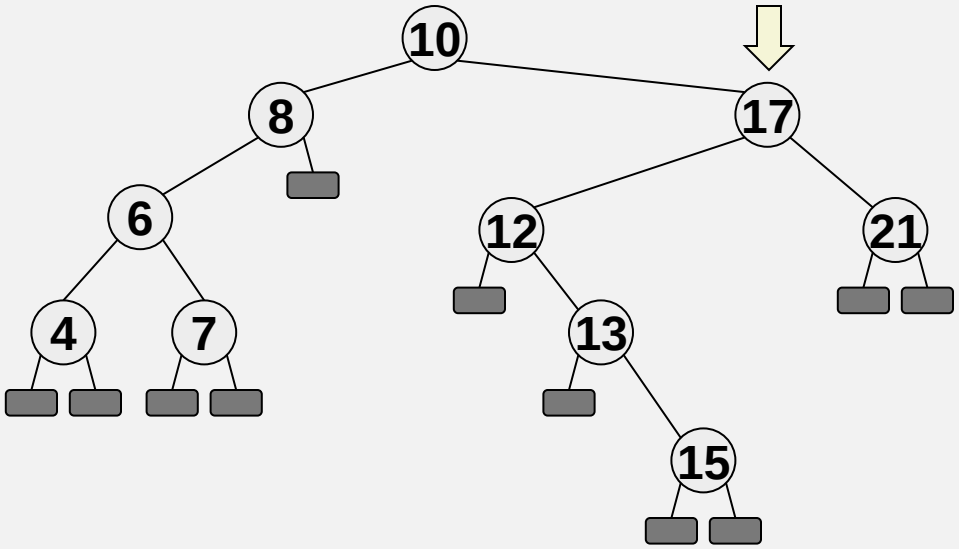


Εισαγωγή στη ρίζα

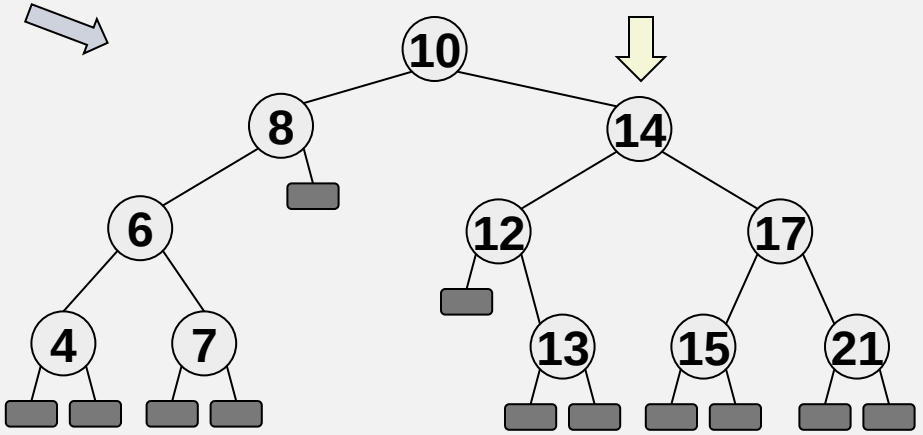


Εισαγωγή στη ρίζα

Εισαγωγή 14

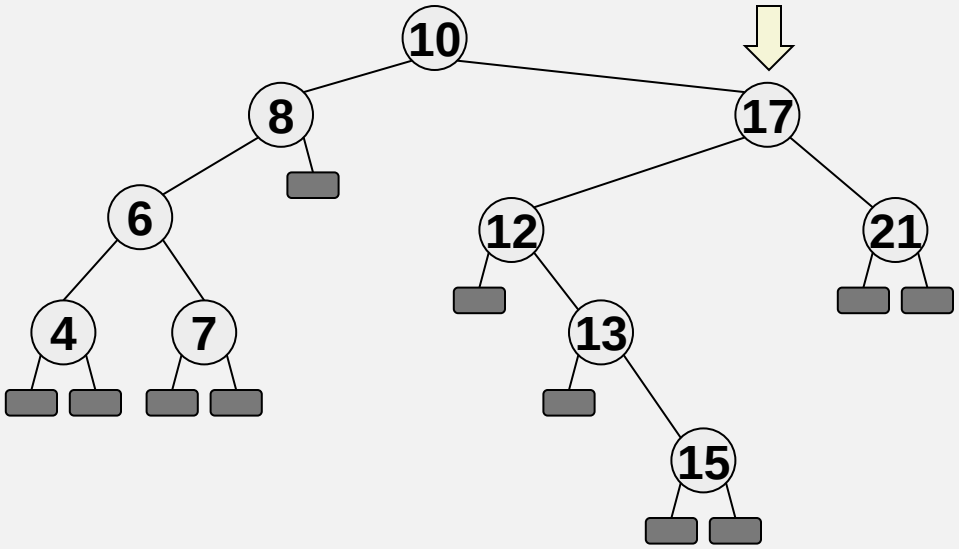


με πιθανότητα 1/6

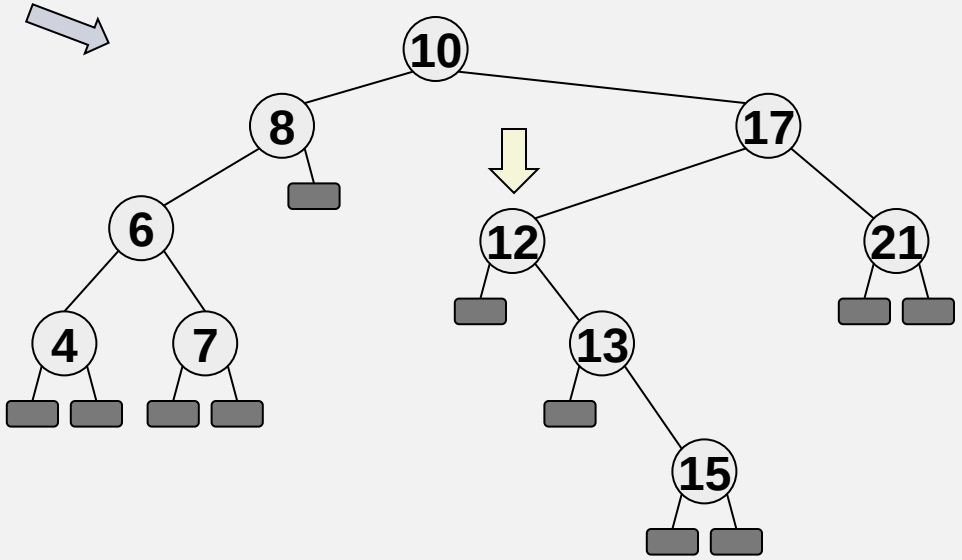


Εισαγωγή στη ρίζα

Εισαγωγή 14

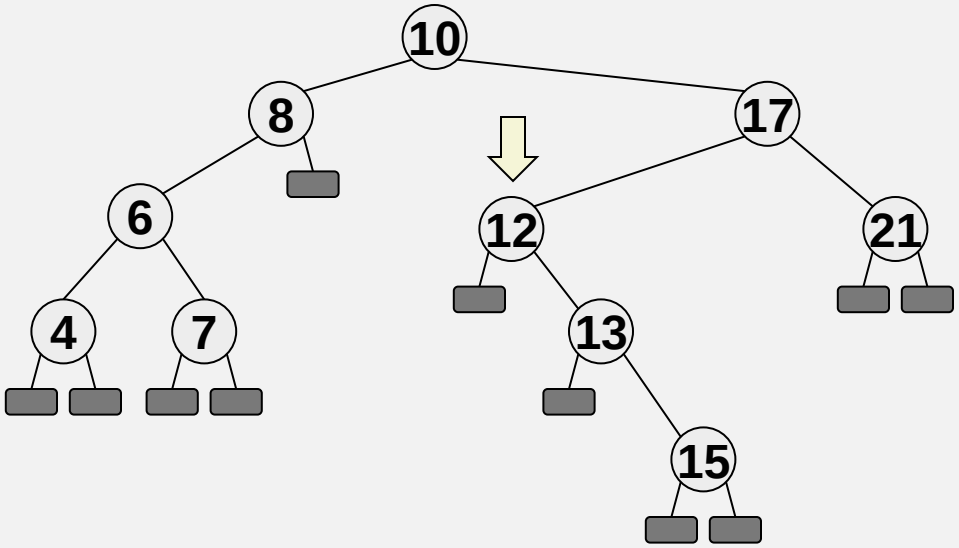


διαφορετικά

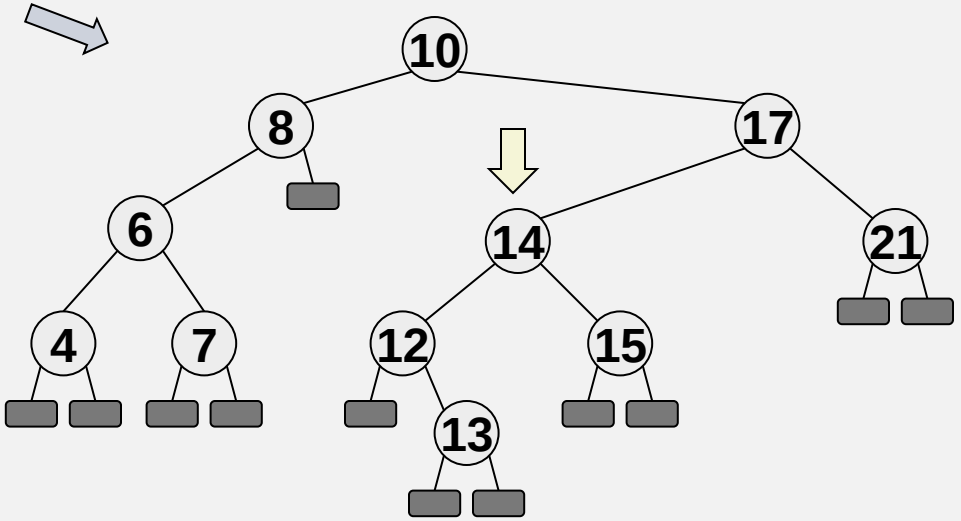


Εισαγωγή στη ρίζα

Εισαγωγή 14

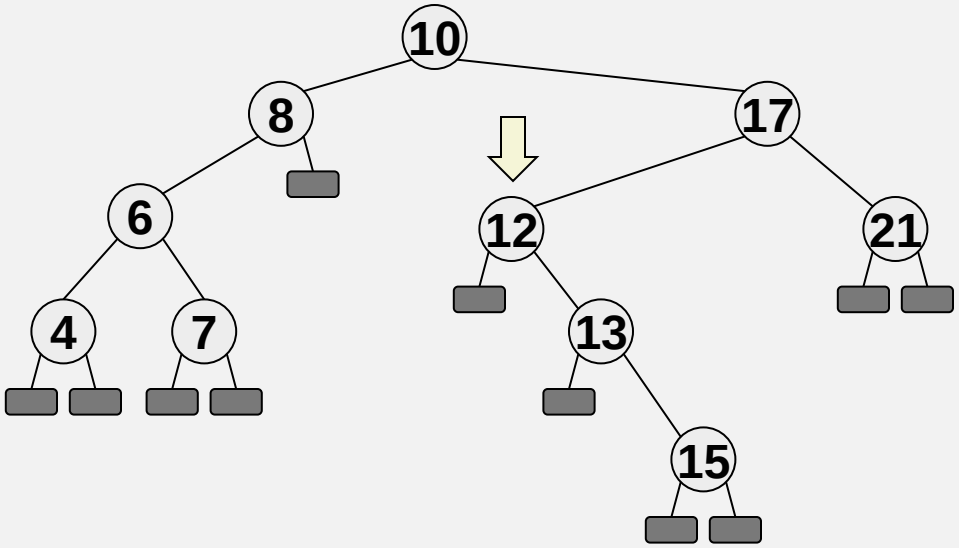


με πιθανότητα 1/4

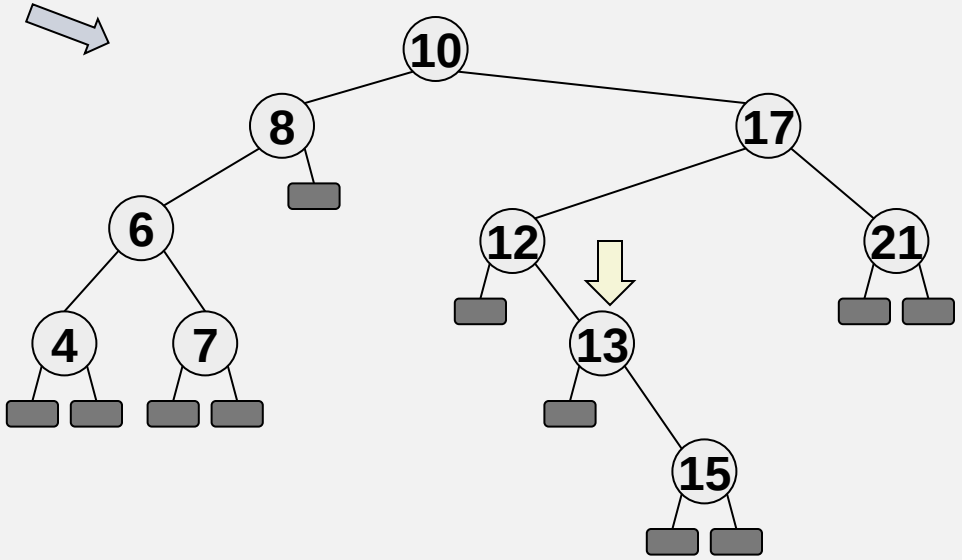


Εισαγωγή στη ρίζα

Εισαγωγή 14

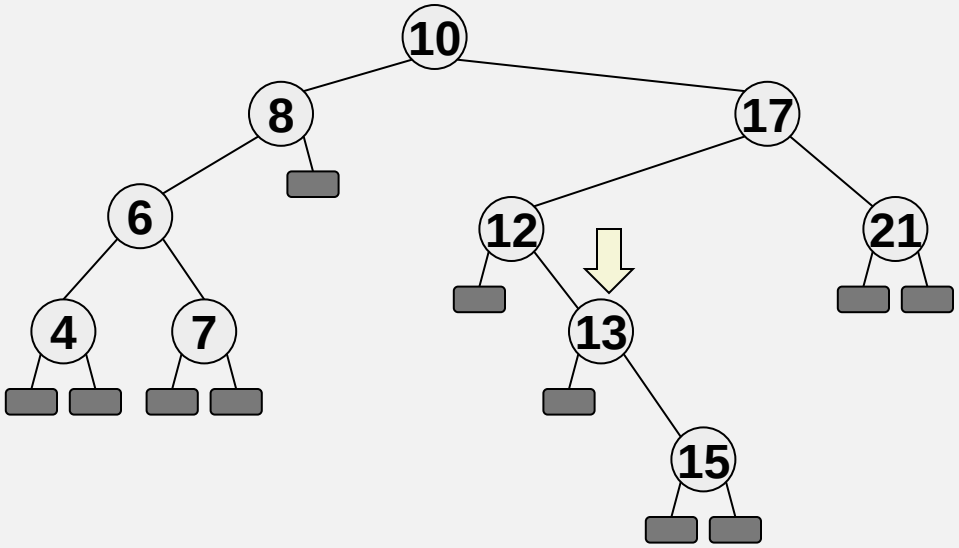


διαφορετικά

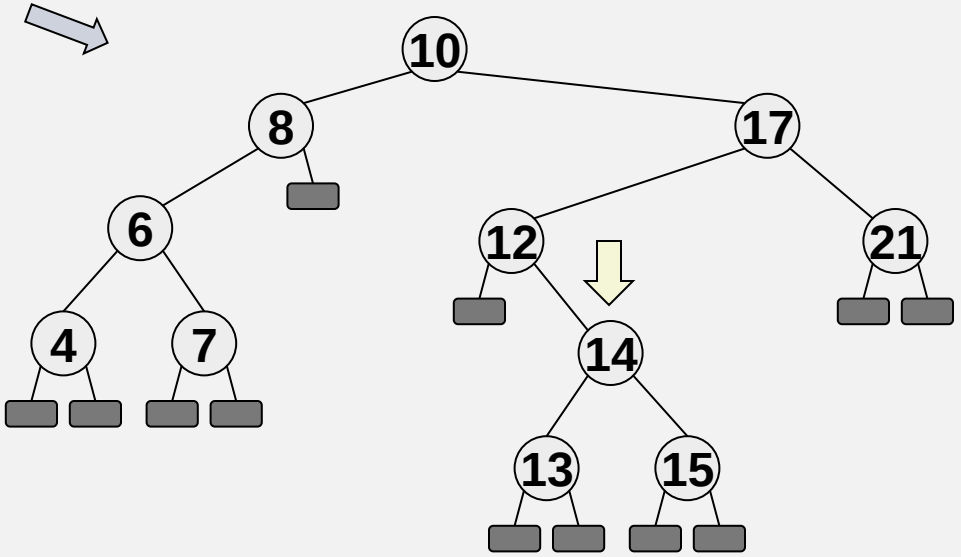


Εισαγωγή στη ρίζα

Εισαγωγή 14

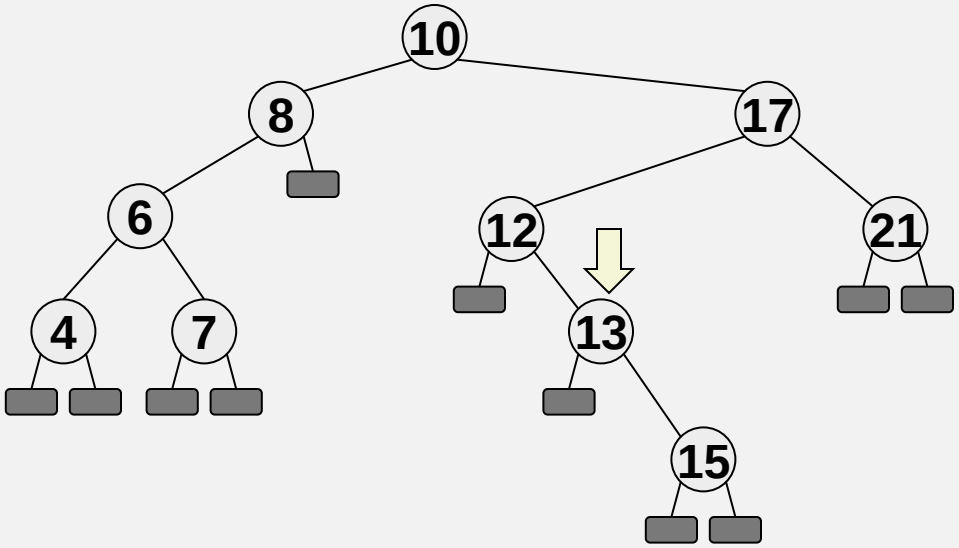


με πιθανότητα 1/3

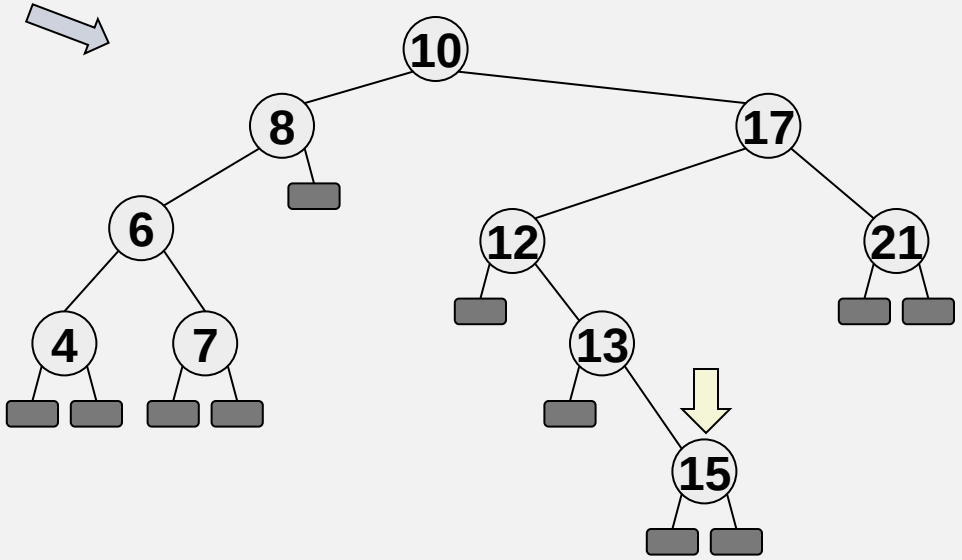


Εισαγωγή στη ρίζα

Εισαγωγή 14

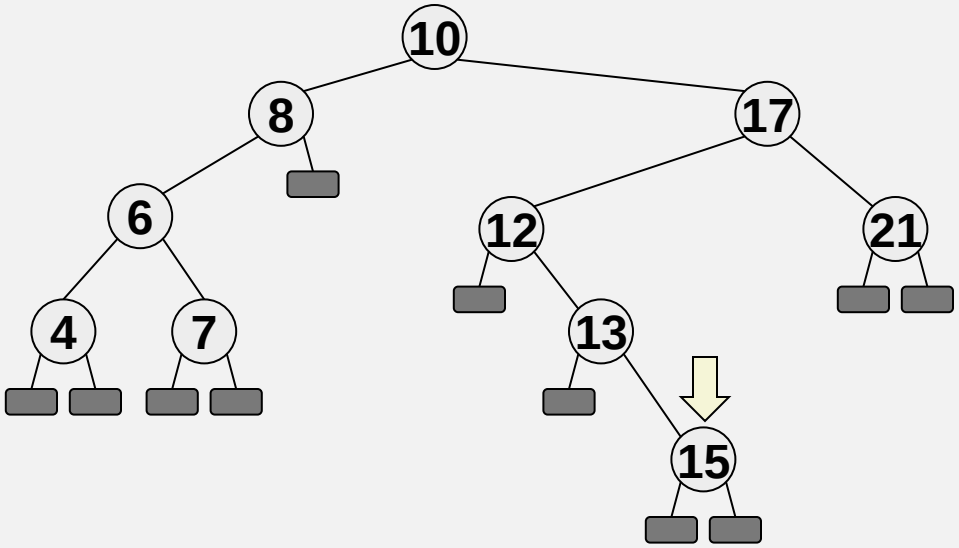


διαφορετικά

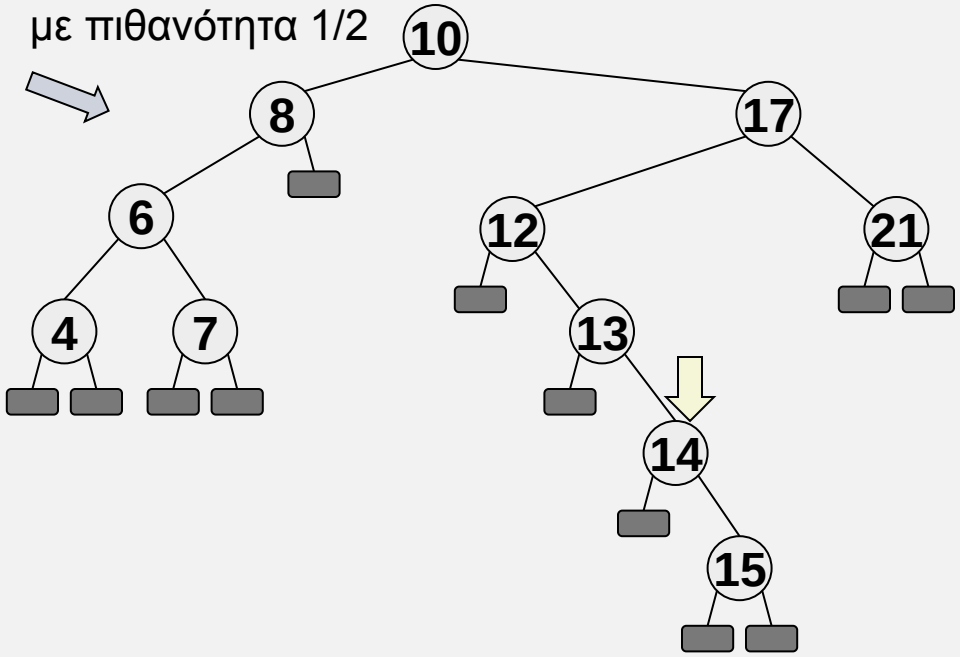


Εισαγωγή στη ρίζα

Εισαγωγή 14

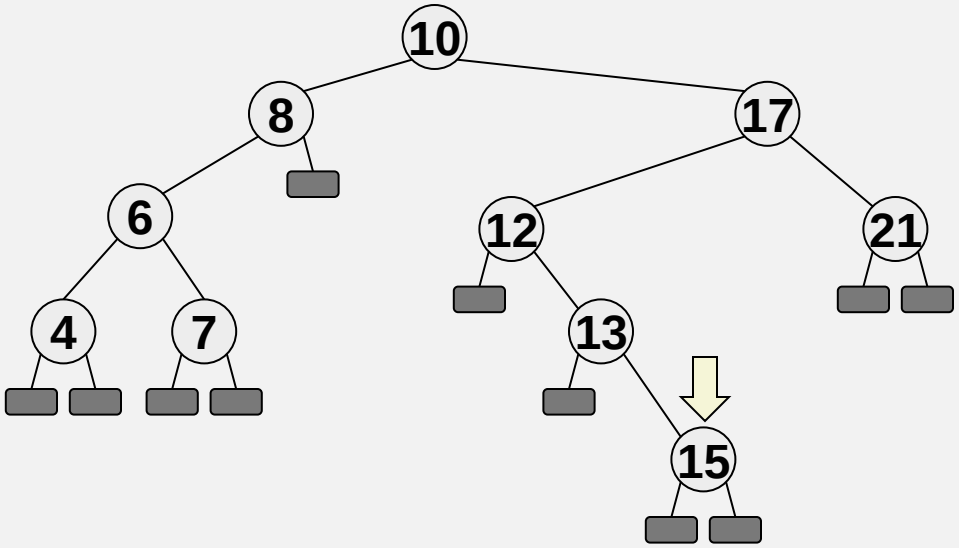


με πιθανότητα 1/2

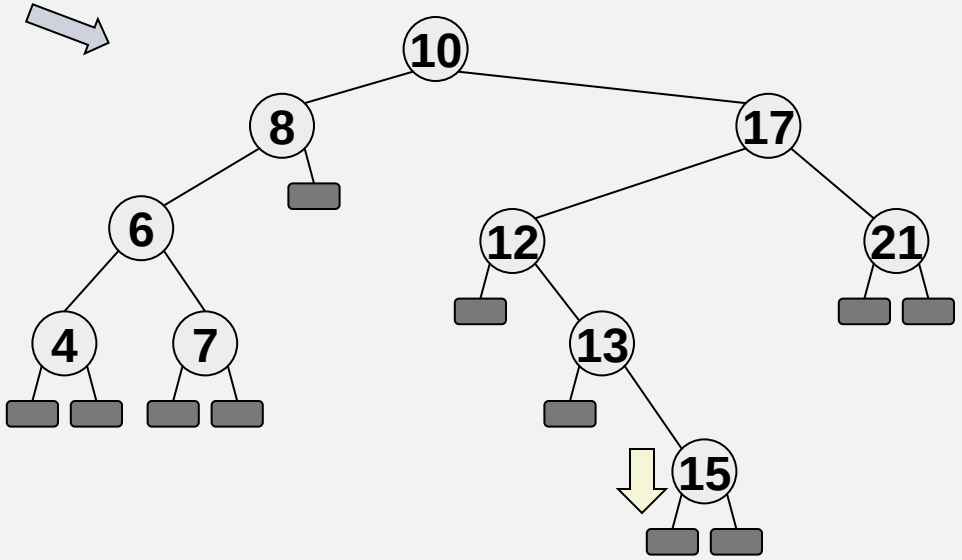


Εισαγωγή στη ρίζα

Εισαγωγή 14

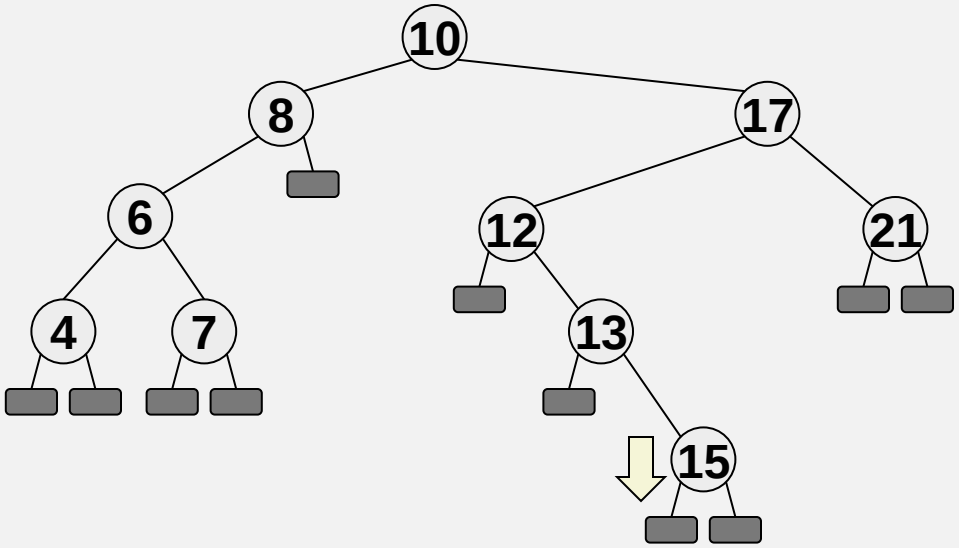


διαφορετικά

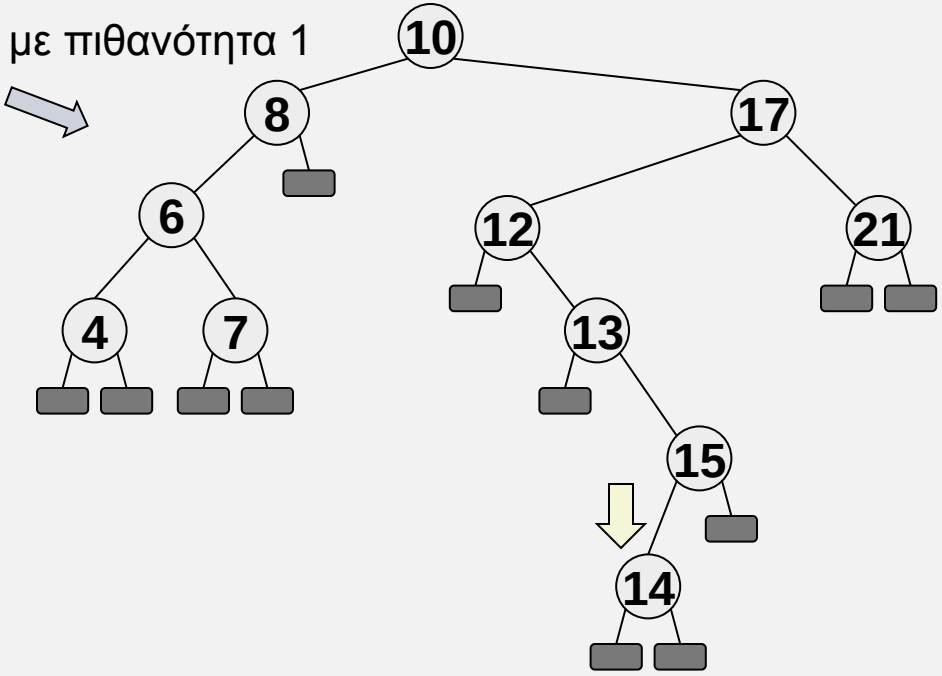


Εισαγωγή στη ρίζα

Εισαγωγή 14



με πιθανότητα 1



Τυχαιοποιημένα δένδρα

Εισαγωγή ενός νέου στοιχείου x σε δένδρο με N στοιχεία

1. Εκτελούμε τον αλγόριθμο εισαγωγής στη ρίζα με πιθανότητα $\frac{1}{N+1}$
2. Διαφορετικά εισάγουμε αναδρομικά το νέο στοιχείο στο κατάλληλο υποδένδρο:
   - Αν το x < κλειδί της ρίζας καλούμε αναδρομικά την εισαγωγή για το αριστερό υποδένδρο
   - Αν το x > κλειδί της ρίζας καλούμε αναδρομικά την εισαγωγή για το δεξιό υποδένδρο

Ιδιότητα: Η κατασκευή ενός τυχαιοποιημένου ΔΔΑ ισοδυναμεί με την κατασκευή ενός καθιερωμένου ΔΔΑ από τυχαίο αρχικό συνδυασμό κλειδιών. Για την κατασκευή ενός τυχαιοποιημένου ΔΔΑ με N στοιχεία απαιτούνται περίπου $2N \ln N$ συγκρίσεις. Μια αναζήτηση κάνει περίπου $2 \ln N$ συγκρίσεις.

Ιδιότητα: Η πιθανότητα το κόστος κατασκευής ενός τυχαιοποιημένου ΔΔΑ να είναι μεγαλύτερο από τη μέση τιμή κατά ένα παράγοντα α είναι $< e^{-\alpha}$