

# Διαδικά Δέντρα Αναζήτησης

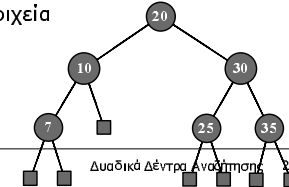
Δημήτρης Φωτάκης

Τμήμα Μηχανικών Πληροφοριακών και Επικοινωνιακών Συστημάτων

Πανεπιστήμιο Αιγαίου

# Διαδικά Δέντρα Αναζήτησης

- Διαδικά Δέντρα με ρίζα.
  - Κάθε εσωτερικός κόμβος περιέχει στοιχείο (αριθμό) και έχει δύο παιδιά.
  - NULL-φύλλα (εξωτερικοί κόμβοι): τερματικοί (NULL) δείκτες χωρίς στοιχεία.
  - Για κάθε κόμβο  $u$ , κόμβο  $left$  στο αριστερό υποδένδρο του  $u$ , και κόμβο  $right$  στο δεξιό υποδένδρο του  $u$ ,
 
$$\text{στοιχείο}(left) \leq \text{στοιχείο}(u) \leq \text{στοιχείο}(right)$$
- Inorder διέλευση τυπώνει στοιχεία σε αύξουσα σειρά.
  - Απόδειξη με επαγωγή.
- Δομή ΔΔΑ επιτρέπει εύκολη αναζήτηση!

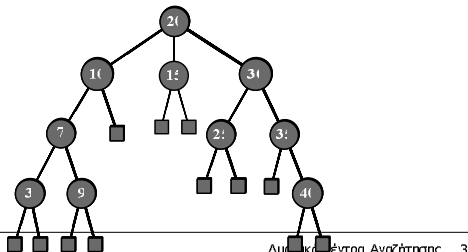


Δομές Δεδομένων

Διαδικά Δέντρα Αναζήτησης

# Διαδικά Δέντρα Αναζήτησης

- Διαδικά Δέντρα με ρίζα.
  - Για κάθε κόμβο  $u$ , κάθε κόμβο  $left$  στο αριστερό υποδένδρο του  $u$ , και κάθε κόμβο  $right$  στο δεξιό υποδένδρο του  $u$ ,
 
$$\text{στοιχείο}(left) \leq \text{στοιχείο}(u) \leq \text{στοιχείο}(right)$$



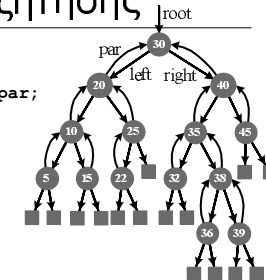
Δομές Δεδομένων

Διαδικά Δέντρα Αναζήτησης 3

# Διαδικά Δέντρα Αναζήτησης

```
typedef struct _tnode {
    struct _tnode *left, *right, *par;
    int key;
} tnode;
tnode root = NULL;
```

- Λειτουργίες:
  - search(x), insert(x), delete(x)
  - min() / max()
  - succ() / pred()

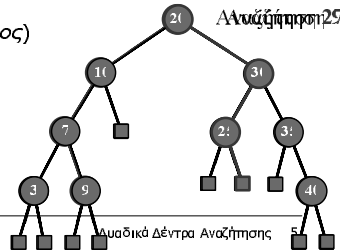


Δομές Δεδομένων

Διαδικά Δέντρα Αναζήτησης 4

## Αναζήτηση

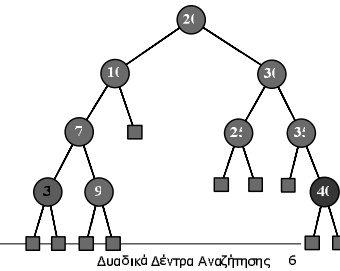
- Ακολουθούμε μονοπάτι από ρίζα προς φύλλο:
  - Αν  $x =$  στοιχείο κόμβου, εύρεση
  - Αν  $x <$  στοιχείο κόμβου, πηγαίνουμε αριστερά
  - Αν  $x >$  στοιχείο κόμβου, πηγαίνουμε δεξιά
  - Ορθότητα: ιδιότητα ΔΔΑ
  - Χρόνος εκτέλεσης:  $O(\text{ύψος})$



Δομές Δεδομένων

## Μέγιστο / Ελάχιστο

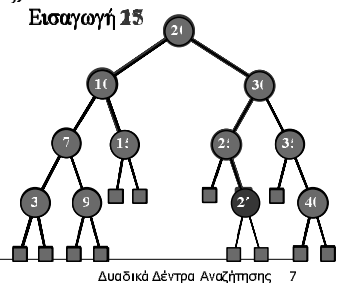
- Μέγιστο : δεξιότατος (rightmost) κόμβος.
- Ελάχιστο : αριστερότατος (leftmost) κόμβος.
- Χρόνος εκτέλεσης:  $O(\text{ύψος})$ .



Δομές Δεδομένων

## Εισαγωγή

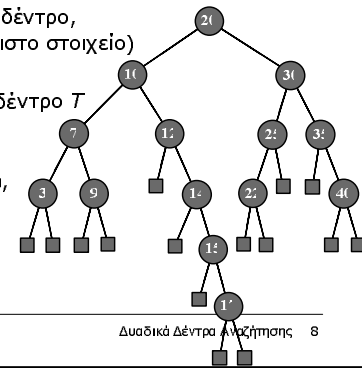
- ... στη θέση NULL-φύλλου που καταλήγει αποτυχημένη αναζήτηση στοιχείου.
- Χρόνος εκτέλεσης:  $O(\text{ύψος})$ .
- Διατηρείται ιδιότητα ΔΔΑ.



Δομές Δεδομένων

## Επόμενος

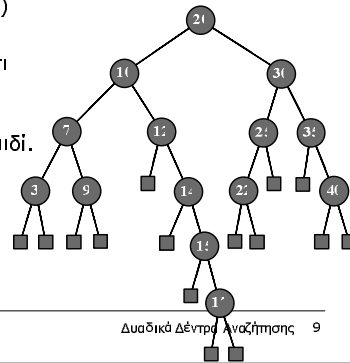
- Επόμενος κόμβος  $p$  στην inorder διέλευση.
- Αν  $p$  έχει (μη-κενό) δεξι υποδέντρο, αριστερότατος κόμβος (ελάχιστο στοιχείο) δεξιού υποδένδρου.
- Διαφορετικά, ψηλότερο υποδέντρο  $T$  για το οποίο  $p$  δεξιότατος κόμβος (μέγιστο στοιχείο).
  - Αν ρίζα  $T$  είναι ρίζα δέντρου,  $p$  τελευταίος κόμβος.
  - Διαφορετικά, ρίζα  $T$  είναι αριστερό παιδί κόμβου  $q$ .
  - Κόμβος  $q$  είναι επόμενος  $p$ .



Δομές Δεδομένων

## Επόμενος

- Αν  $p$  έχει (μη-κενό) δεξί υποδέντρο, αριστερότατος κόμβος (ελάχιστο στοιχείο) δεξιού υποδένδρου.
- Διαφορετικά, στο μονοπάτι από  $p$  προς ρίζα, πρώτος πατέρας επισκεπτόμενος από δείκτη σε αριστερό παιδί.



Δομές Δεδομένων

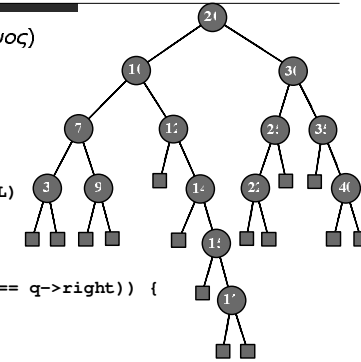
Διαδικά Δέντρα Αναζήτησης 9

## Επόμενος

- Χρόνος εκτέλεσης:  $O(\text{ύψος})$

```

tnode *succ(tnode *p) {
    if (p->right != NULL) {
        p = p->right;
        while (p->left != NULL)
            p = p->left;
        return(p);
    }
    tnode *q = p->par;
    while ((q != NULL) && (p == q->right)) {
        p = q; q = p->par;
    }
    return(q);
}
    
```

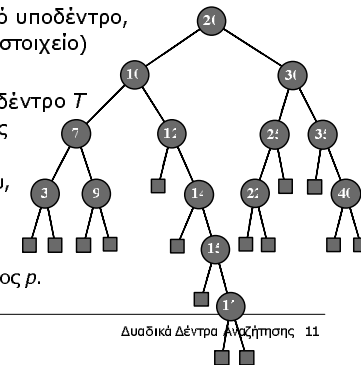


Δομές Δεδομένων

Διαδικά Δέντρα Αναζήτησης 10

## Προηγούμενος

- Προηγούμενος κόμβου  $p$  στην inorder διέλευση.
- Αν  $p$  έχει (μη-κενό) αριστερό υποδέντρο, δεξιότατος κόμβος (μέγιστο στοιχείο) αριστερού υποδένδρου.
- Διαφορετικά, ψηλότερο υποδέντρο  $T$  για το οποίο  $p$  αριστερότατος κόμβος (ελάχιστο στοιχείο).
  - Αν ρίζα  $T$  είναι ρίζα δέντρου,  $p$  πρώτος κόμβος.
  - Διαφορετικά, ρίζα  $T$  είναι δεξί παιδί κόμβου  $q$ .
  - Κόμβος  $q$  είναι προηγούμενος  $p$ .

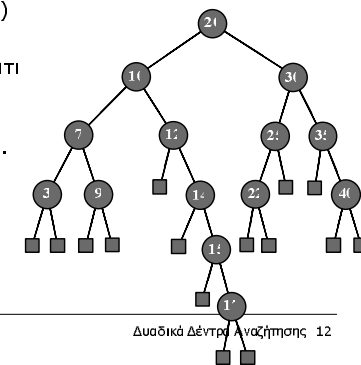


Δομές Δεδομένων

Διαδικά Δέντρα Αναζήτησης 11

## Προηγούμενος

- Αν  $p$  έχει (μη-κενό) αριστερό υποδέντρο, δεξιότατος κόμβος (μέγιστο στοιχείο) αριστερού υποδένδρου.
- Διαφορετικά, στο μονοπάτι από  $p$  προς ρίζα, πρώτος πατέρας επισκεπτόμενος από δείκτη σε δεξιά παιδί.



Δομές Δεδομένων

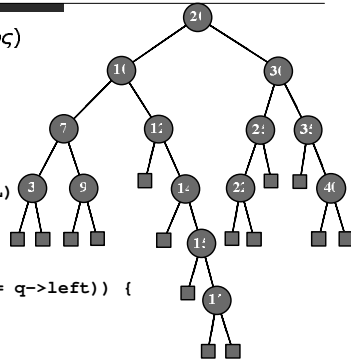
Διαδικά Δέντρα Αναζήτησης 12

## Προηγούμενος

- Χρόνος εκτέλεσης:  $O(\text{ύψος})$

```

tnode *pred(tnode *p) {
    if (p->left != NULL) {
        p = p->left;
        while (p->right != NULL)
            p = p->right;
        return(p);
    }
    tnode *q = p->par;
    while ((q != NULL) && (p == q->left)) {
        p = q; q = p->par;
    }
    return(q);
}
    
```

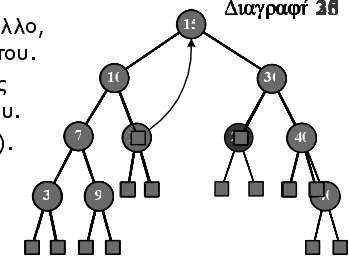


Δομές Δεδομένων

Διαδικά Δέντρα Αναζήτησης 13

## Διαγραφή

- Εντοπισμός κόμβου  $p$  με αναζήτηση.
- Αν  $p$  έχει δύο παιδιά NULL-φύλλα, διαγράφεται.
- Αν  $p$  έχει ένα παιδί NULL-φύλλο, άλλο παιδί του  $p$  στη θέση του.
- Διαφορετικά, προηγούμενος (ή επόμενος)  $p$  στη θέση του.
- Χρόνος εκτέλεσης:  $O(\text{ύψος})$ .
- Διατηρείται ιδιότητα ΔΔΑ.



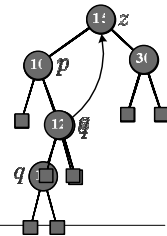
Δομές Δεδομένων

Διαδικά Δέντρα Αναζήτησης 14

## Διαγραφή

```

int delete(int x) {
    tnode *p, *q, *z = search(x);
    if (z == NULL) return(NOT_FOUND);
    if (z->left && z->right) { // 2 παιδιά -> 1 παιδί
        q = pred(z); // δεξιότατο αριστερού υποδέντρου
        z->key = q->key; z = q;
    }
    if (z->left) q = z->left;
    else q = z->right;
    p = z->par;
    if (q) q->par = p;
    if (z == root) root = q;
    else if (z == p->left) p->left = q;
    else p->right = q;
    free(z); return(OK);
}
    
```

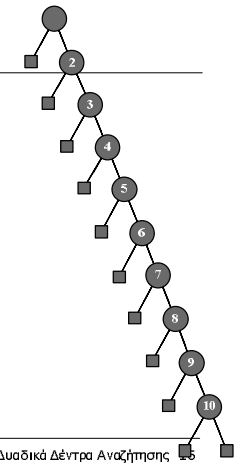


Δομές Δεδομένων

Διαδικά Δέντρα Αναζήτησης 15

## Ύψος - Απόδοση

- Χειρότερη περίπτωση :  
Γραμμικό ύψος –  $\Theta(n)$ .
  - Εισαγωγή  $1, 2, \dots, n$
  - Εισαγωγή  $n, n-1, \dots, 2, 1$
  - Εκφυλίζεται σε γραμμική λίστα.
  - Γραμμικό κόστος λειτουργιών.
- Καλύτερη περίπτωση :  
Λογαριθμικό ύψος –  $O(\log n)$ .
- Εισαγωγή τυχαίων στοιχείων :  
Λογαριθμικό ύψος –  $O(\log n)$ .
- Ζυγισμένα δέντρα διορθώνουν χειρότερη περίπτωση.



Δομές Δεδομένων

Διαδικά Δέντρα Αναζήτησης 16

## Ασκήσεις

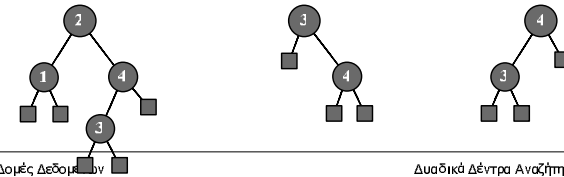
- ΔΔΑ περιέχει κάποιους αριθμούς από 1 έως 1000. Αναζητούμε το 363. Ποια δεν μπορεί να προκύψουν;
  - 2, 252, 401, 398, 330, 344, 397, 363
  - 924, 220, 911, 244, 898, 258, 362, 363
  - 925, 202, 911, 240, 912, 245, 363
  - 2, 399, 387, 219, 266, 282, 381, 278, 363
  - 935, 278, 347, 621, 299, 392, 358, 363
- Έστω κόμβος  $p$  με δύο παιδιά που δεν είναι NULL-φύλλα.  $\text{succ}(p)$  /  $\text{pred}(p)$  έχει αριστερό / δεξί παιδί NULL-φύλλο.
- Επιλογή ( $k$ -οστό μικρότερο στοιχείο) σε ΔΔΑ;
  - Πλήθος στοιχείων στο αριστερό υποδέντρο + 1 (για τρέχον στοιχείο – ρίζα υποδέντρου). Πεδίο LeftSize.

Δομές Δεδομένων

Διαδικά Δέντρα Αναζήτησης 17

## Ασκήσεις

- Ταξινόμηση με εισαγωγή σε ΔΔΑ. Χρόνος εκτέλεσης;
- Εισαγωγή δεν είναι αντιμεταθετική.
  - Εισαγωγή 1, 2, 3.
  - Εισαγωγή 2, 1, 3.
- Διαγραφή δεν είναι αντιμεταθετική.
  - Σβήνω 2 και μετά 1.
  - Σβήνω 1 και μετά 2.

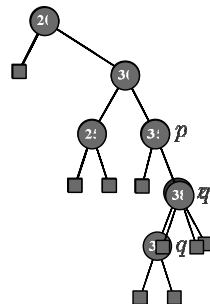


Δομές Δεδομένων

Διαδικά Δέντρα Αναζήτησης 18

## Ασκήσεις

- Διαγραφή μέγιστου (ελάχιστου).
  - Δεξιότατος κόμβος έχει δεξί παιδί NULL-φύλλο.
  - Είτε ρίζα είτε δεξί παιδί.



Δομές Δεδομένων

Διαδικά Δέντρα Αναζήτησης 19