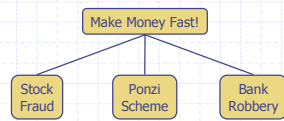


Δέντρα



4/11/2005 4:49 PM

Δέντρα

1

Κύρια σημεία για μελέτη

- ◆ Ο ΑΤΔ του δέντρου (§2.3.1)
- ◆ Preorder και postorder διασχίσεις (§2.3.2)
- ◆ Ο ΑΤΔ του δυαδικού δέντρου (§2.3.3)
- ◆ Inorder διάσχιση (§2.3.3)
- ◆ Η διάσχιση του Euler (§2.3.3)
- ◆ Template method pattern
- ◆ Δομές δεδομένων για δέντρα (§2.3.4)
- ◆ Εφαρμογές σε Java (<http://jdsi.org>)

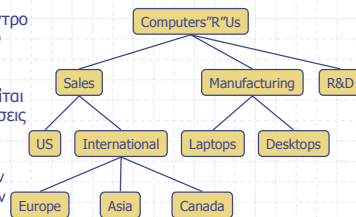
4/11/2005 4:49 PM

Δέντρα

2

Τι είναι δέντρο

- ◆ Στην επιστήμη υπολογιστών, το δέντρο είναι ένα αφηρημένο μοντέλο ιεραρχικής δόμησης.
- ◆ Ένα δέντρο αποτελείται από κόμβους με σχέσεις πατέρα παιδιού.
- ◆ Εφαρμογές σε:
 - Χάρτες οργανισμών
 - Συστήματα αρχείων
 - Προγραμμαστικά περιβάλλοντα



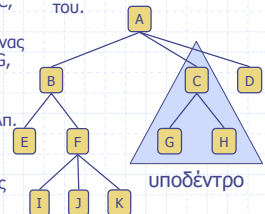
4/11/2005 4:49 PM

Δέντρα

3

Ορολογία δέντρων

- ◆ Κόμβος Ρίζα: ένας κόμβος χωρίς πατέρα (A)
- ◆ Εσωτερικός κόμβος: Ένας κόμβος με τουλάχιστον ένα παιδί (A, B, C, F)
- ◆ Εξωτερικός κόμβος(ή φύλλο): ένας κόμβος χωρίς παιδιά (E, I, J, K, G, H, D)
- ◆ Πρόγονοι ενός κόμβου: κόμβος πατέρας, παππούς, προπάππος κλπ.
- ◆ Βάθος κόμβου: ο αριθμός των κόμβων προγόνων.
- ◆ Ύψος δέντρου: το μέγιστο βάθος οποιουδήποτε κόμβου του δέντρου.(3)
- ◆ Απόγονος κόμβου: παιδί, εγγόνι...



4/11/2005 4:49 PM

Δέντρα

4

Ο ΑΤΔ του δέντρου

- Χρησιμοποιούμε θέσεις για να γενικεύσουμε τους κόμβους.
- Γενικές μέθοδοι:
 - integer `size()`
 - boolean `isEmpty()`
 - objectIterator `elements()`
 - positionIterator `positions()`
- Συμπληρωματικές μέθοδοι:
 - position `root()`
 - position `parent(p)`
 - positionIterator `children(p)`
- Μέθοδοι ερωτημάτων:
 - boolean `isInternal(p)`
 - boolean `isExternal(p)`
 - boolean `isRoot(p)`
- Μέθοδοι αναβάθμισης:
 - object `swapElements(p, q)`
 - object `replaceElement(p, o)`
- Πρόσθετοι μέθοδοι αναβάθμισης μπορεί να ορίζονται από τις δομές δεδομένων που υλοποιούν τον ΑΤΔ του δέντρου.

4/11/2005 4:49 PM

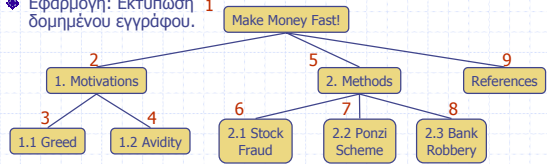
Δέντρο

5

Preorder διάσχιση

- Μια διάσχιση επισκέφεται τους κόμβους του δέντρου με συστηματικό τρόπο.
- Σε μια preorder διάσχιση, η επίσκεψη σε έναν κόμβο γίνεται πριν από την επίσκεψη στους απογόνους του.
- Εφαρμογή: Εκτύπωση δομημένου εγγράφου.

Algorithm *preOrder(v)*
visit(v)
for each child *w* of *v*
preorder(w)



4/11/2005 4:49 PM

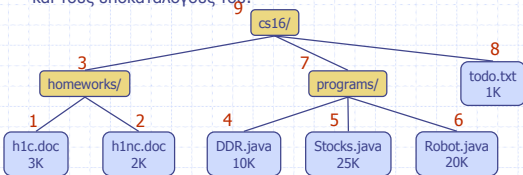
Δέντρο

6

Postorder διάσχιση

- Σε μια postorder διάσχιση, η επίσκεψη σε έναν κόμβο γίνεται μετά από την επίσκεψη στους απογόνους του.
- Εφαρμογή: υπολογισμός του χώρου που χρησιμοποιείται από αρχεία σε έναν κατάλογο και τους υποκαταλόγους του.

Algorithm *postOrder(v)*
for each child *w* of *v*
postOrder(w)
visit(v)



4/11/2005 4:49 PM

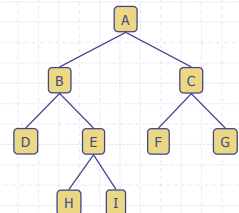
Δέντρο

7

Διαδικό δέντρο

- Διαδικό είναι ένα δέντρο με τις ακόλουθες ιδιότητες:
 - Κάθε εσωτερικός κόμβος έχει δύο παιδιά
 - Τα παιδιά ενός κόμβου αποτελούν ένα διατεταγμένο δέντρο.
- Αποκαλούμε τα παιδιά ενός εσωτερικού κόμβου αριστερό και δεξιό παιδί.
- Εναλλακτικός αναδρομικός ορισμός: ένα διαδικό δέντρο είναι είτε:
 - Ένα δέντρο αποτελούμενο από ένα μόνο κόμβο είτε
 - Ένα δέντρο του οποίου η ρίζα έχει ένα διατεταγμένο ζευγάρι από παιδιά, το καθένα εκ των οποίων είναι διαδικό δέντρο.

- Εφαρμογές:
 - Αριθμητικές εκφράσεις
 - Διεργασίες αποφάσεων
 - Αναζητήσεις



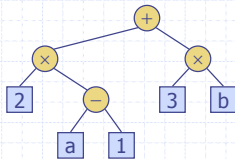
4/11/2005 4:49 PM

Δέντρο

8

Δέντρα αριθμητικών εκφράσεων

- ◆ Σε ένα δυαδικό δέντρο που σχετίζεται με μια αριθμητική έκφραση,
 - Οι εσωτερικοί κόμβοι είναι: τελεστές
 - Οι εξωτερικοί κόμβοι είναι: τελεστέοι
- ◆ Παράδειγμα: το δέντρο αριθμητικής έκφρασης για την έκφραση $(2 \times (a - 1) + (3 \times b))$ είναι το παρακάτω.



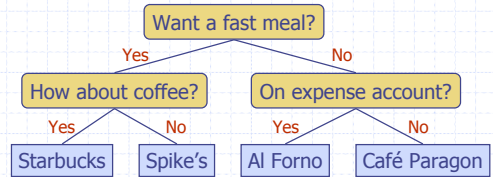
4/11/2005 4:49 PM

Δέντρα

9

Δέντρα Απόφασης (decision trees)

- ◆ Σε ένα δυαδικό δέντρο το οποίο σχετίζεται με μια διαδικασία απόφασης
 - Οι εσωτερικοί κόμβοι είναι: ερωτήσεις με απάντηση ναι ή όχι
 - Οι εξωτερικοί κόμβοι είναι: αποφάσεις
- ◆ Παράδειγμα: απόφαση για δείπνο



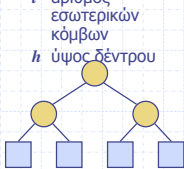
4/11/2005 4:49 PM

Δέντρα

10

Ιδιότητες των δυαδικών δέντρων

- ◆ Συμβολισμός
 - n αριθμός κόμβων
 - e αριθμός εξωτερικών κόμβων
 - i αριθμός εσωτερικών κόμβων
 - h ύψος δέντρου
- ◆ Ιδιότητες που ισχύουν:
 - $e = i + 1$
 - $n = 2e - 1$
 - $h \leq i$
 - $h \leq (n - 1)/2$
 - $e \leq 2^h$
 - $h \geq \log_2 e$
 - $h \geq \log_2 (n + 1) - 1$



4/11/2005 4:49 PM

Δέντρα

11

Ο ΑΤΔ του δυαδικού δέντρου

- ◆ Ο ΑΤΔ του δυαδικού δέντρου αποτελεί προέκταση του ΑΤΔ του δέντρου, κληρονομώντας όλες τις μεθόδους του.
- ◆ Επιπλέον μέθοδοι:
 - position `leftChild(p)`
 - position `rightChild(p)`
 - position `sibling(p)`
- ◆ Μέθοδοι αναβάθμισης μπορεί να ορίζονται από τις δομές δεδομένων που υλοποιούν τον ΑΤΔ του δυαδικού δέντρου.

4/11/2005 4:49 PM

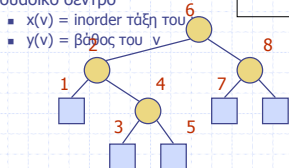
Δέντρα

12

Inorder διάσχιση

- ◆ Σε μια Inorder διάσχιση η επίσκεψη στον κόμβο γίνεται μετά την επίσκεψη στο αριστερό υποδέντρο και πριν την επίσκεψη στο δεξί υποδέντρο.

- ◆ Εφαρμογή: Ζωγραφίστε ένα δυαδικό δέντρο



Algorithm *inOrder(v)*

```

if isInternal (v)
    inOrder (leftChild (v))
    visit(v)
if isInternal (v)
    inOrder (rightChild (v))
    
```

4/11/2005 4:49 PM

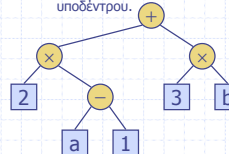
Δέντρα

13

Εκτύπωση αριθμητικών εκφράσεων

- ◆ Εξειδίκευση μιας inorder διάσχισης

- Εκτύπωσε τον τελεστή ή τον τελεστέο όταν επισκέφτεσαι τον κόμβο
- εκτύπωσε "(" πριν την διάσχιση του αριστερού υποδέντρου
- εκτύπωσε ")" μετά την διάσχιση του δεξιού υποδέντρου.



Algorithm *printExpression(v)*

```

if isInternal (v)
    print("(")
    inOrder (leftChild (v))
    print(v.element ())
if isInternal (v)
    inOrder (rightChild (v))
    print(")")
    
```

$((2 \times (a - 1)) + (3 \times b))$

4/11/2005 4:49 PM

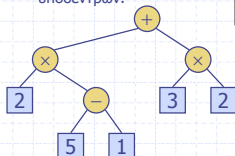
Δέντρα

14

Αξιολόγηση αριθμητικών εκφράσεων

- ◆ Εξειδίκευση της postorder διάσχισης

- Μια αναδρομική μέθοδος επιστρέφει την τιμή ενός υποδέντρου.
- Κατά την επίσκεψη ενός εσωτερικού κόμβου, συνδυάζονται οι τιμές των υποδέντρων.



Algorithm *evalExpr(v)*

```

if isExternal (v)
    return v.element ()
else
    x ← evalExpr(leftChild (v))
    y ← evalExpr(rightChild (v))
    φ ← operator stored at v
    return x φ y
    
```

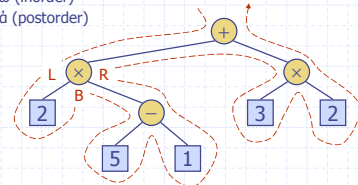
4/11/2005 4:49 PM

Δέντρα

15

Διάσχιση του Euler

- ◆ Γενική διάσχιση ενός δυαδικού δέντρου.
- ◆ Περιλαμβάνει τις διασχίσεις preorder, postorder και inorder
- ◆ Γίνεται κυκλική διάσχιση του κόμβου και η επίσκεψη σε κάθε κόμβο γίνεται τρεις φορές:
 - Στα αριστερά (preorder)
 - Από κάτω (inorder)
 - Στα δεξιά (postorder)



4/11/2005 4:49 PM

Δέντρα

16

Template Method Pattern

- ◆ Γενικός αλγόριθμος ο οποίος μπορεί να εξειδικευθεί με τον επαναρισμό συσκευασμένων βημάτων
- ◆ Υλοποιείται μέσω μιας Java class
- ◆ Οι μέθοδοι επίσκεψης μπορούν να επαναριστούν από υποκλάσεις
- ◆ Template method `eulerTour`
 - Καλείται αναδρομικά στο δεξί και στο αριστερό παιδί
 - Ένα `Result` αντικείμενο με πεδία `leftResult`, `rightResult` και `finalResult` ανιχνεύει το output της κλήσης της αναδρομικής συνάρτησης `eulerTour`

```
public abstract class EulerTour {
    protected BinaryTree tree;
    protected void visitExternal(Position p, Result r) {}
    protected void visitLeft(Position p, Result r) {}
    protected void visitBelow(Position p, Result r) {}
    protected void visitRight(Position p, Result r) {}
    protected Object eulerTour(Position p) {
        Result r = new Result();
        if tree.isExternal(p) { visitExternal(p, r); }
        else {
            visitLeft(p, r);
            r.leftResult = eulerTour(tree.leftChild(p));
            visitBelow(p, r);
            r.rightResult = eulerTour(tree.rightChild(p));
            visitRight(p, r);
            return r.finalResult;
        }
    }
}
```

4/11/2005 4:49 PM

Δέντρα

17

Εξειδικεύσεις του EulerTour

- ◆ Δείχνουμε πώς να εξειδικεύσουμε την class `EulerTour` για αξιολόγηση αριθμητικών εκφράσεων.
- ◆ Υποθέσεις
 - Οι εξωτερικοί κόμβοι αποθηκεύουν ακέραια αντικείμενα
 - Οι εσωτερικοί κόμβοι αποθηκεύουν `Operator` αντικείμενα που υποστηρίζουν μεθόδους `operation (Integer, Integer)`

```
public class EvaluateExpression
    extends EulerTour {
    protected void visitExternal(Position p, Result r) {
        r.finalResult = (Integer) p.element();
    }
    protected void visitRight(Position p, Result r) {
        Operator op = (Operator) p.element();
        r.finalResult = op.operation(
            (Integer) r.leftResult,
            (Integer) r.rightResult
        );
    }
    ...
}
```

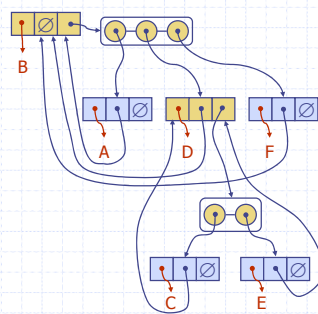
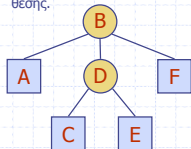
4/11/2005 4:49 PM

Δέντρα

18

Δομές δεδομένων για δέντρα

- ◆ Ένας κόμβος αναπαριστάται από ένα αντικείμενο στο οποίο αποθηκεύονται
 - Ένα στοιχείο
 - Ο κόμβος πατέρας
 - Η ακολουθία από κόμβους παιδιά
- ◆ Αντικείμενα κόμβων υλοποιούν τον ΑΤΔ της θέσης.



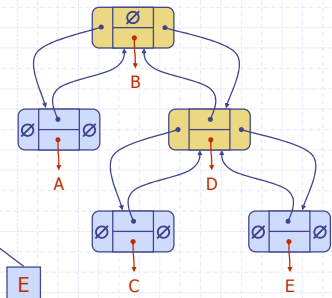
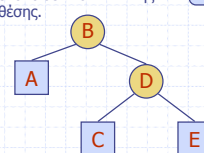
4/11/2005 4:49 PM

Δέντρα

19

Δομές δεδομένων για δυαδικά δέντρα

- ◆ Ένας κόμβος αναπαριστάται από ένα αντικείμενο στο οποίο αποθηκεύονται
 - Ένα στοιχείο
 - Ο κόμβος πατέρας
 - Αριστερός κόμβος παιδί
 - Δεξιός κόμβος παιδί
- ◆ Αντικείμενα κόμβων υλοποιούν τον ΑΤΔ της θέσης.



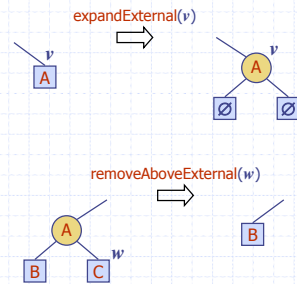
4/11/2005 4:49 PM

Δέντρα

20

Υλοποίηση σε JAVA

- ◆ Interface για δέντρο
- ◆ Interface για δυαδικό δέντρο που επεκτείνει το δέντρο
- ◆ Κλάσεις υλοποιούν το δέντρο και το δυαδικό δέντρο και παρέχουν
 - Constructors
 - Μεθόδους αναβάθμισης
 - Μεθόδους εκτύπωσης
- ◆ Παράδειγμα αναβαθμίσεων για δυαδικά δέντρα
 - `expandExternal(v)`
 - `removeAboveExternal(w)`



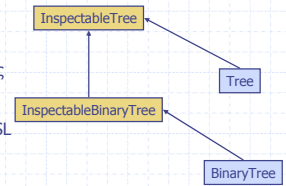
4/11/2005 4:49 PM

Δέντρα

21

Δέντρα σε JDSL

- ◆ Η JDSL είναι μια βιβλιοθήκη για δομές δεδομένων σε Java
- ◆ Η JDSL αναπτύχθηκε στο: Brown's Center for Geometric Computing
- ◆ JDSL documentation and tutorials at <http://jdsl.org>
- ◆ interfaces δέντρων σε JDSL
 - Επιθεωρήσιμο Δυαδικό δέντρο
 - Επιθεωρήσιμο δέντρο
 - Δυαδικό δέντρο
 - Δέντρο
- ◆ Οι επιθεωρήσιμες εκδόσεις δεν έχουν μεθόδους αναβάθμισης
- ◆ Κλάσεις για δέντρα σε JDSL
 - NodeBinaryTree
 - NodeTree



4/11/2005 4:49 PM

Δέντρα

22