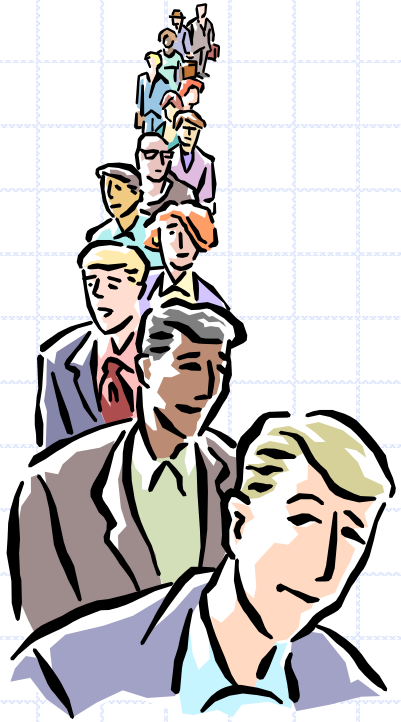
Επαναλήπτες και Ακολουθίες



Επαναλήπτες

- Ένας επαναλήπτης (iterator) αποτελεί αφαίρεση της διαδικασίας σάρωσης μιας συλλογής στοιχείων
- Έχει εναν δρομέα (cursor) που στέκεται μεταξύ δύο στοιχείων της λίστας, ή πριν από το πρώτο ή μετά το τελέυταίο στοιχείο
- Μέθοδοι του ΑΤΔ επαναλήπτη:
  - hasNext(): επιστρέφει true όσο η λίστα δεν είναι κενή και ο δρομέας δεν βρίσκεται μετά από το τελευταίο στοιχείο
  - next(): επιστρέφει το επόμενο στοιχείο
- Επεκτείνει την έννοια της θέσης προσθέτοντας δυνατότητα σάρωσης
- Υλοποίηση με πίνακα ή απλά συνδεδεμένη λίστα

Κλάσεις Επαναληπτών

- Ένας επαναλήπτης τυπικά συνδέεται με μια άλλη δομή δεδομένων, που μπορεί να υλοποιήσει τον ΑΤΔ επανάληψης
- Μπορούμε να επεκτείνουμε τους ΑΤΔ στοίβας, ουράς , διανύσματος. Λίστας και ακολουθίας με την μέθοδο:
  - Iterator<E> iterator(): επιστρέφει εναν επαναλήπτη στα στοιχεία
  - Στην Java, κλάσεις με αυτή τη μέθοδο επεκτείνουν Iterable<E>
- Δύο έννοιες του επαναλήπτη:
  - snapshot: παγώνει τα περιεχόμενα της δομής δεδομένων σε συγκεκριμένη χρονική στιγμή
  - dynamic: ακολουθεί αλλαγές στη δομή δεδομένων
  - Στη Java: ένας επαναλήπτης θα αποτύχει (και θα δώσει εξαίρεση) αν η συλλογή αλλάξει απρόσμενα

Η Επανάληψη For-Each

- Η Java υποστηρίζει έναν απλό τρόπο σάρωσης των στοιχείων μιας επαναλήψιμης κλάσης:
  - for (type name: expression)
    loop_body
  - Για παράδειγμα:

```
List<Integer> values;
int sum=0
for (Integer i : values)
  sum += i;   // boxing/unboxing allows this
```

Υλοποίηση Επαναληπτών

- Βασισμένη σε πίνακες
  - ο πίνακας A από στοιχεία
  - ο δείκτης i που έχει τιμή τον δρομέα
- Βασισμένη σε συνδεδεμένη λίστα
  - διπλά συνδεδεμένη λίστα L που αποθηκεύει τα στοιχεία, με φρουρό την κεφαλή και το τέλος
  - δείκτης p στον κόμβο που περιέχει το τελευταίο στοιχείο που επιστρέφεται (ή την κεφαλή αν αυτός είναι ένας νέος επαναλήπτης).
- Μπορούμε να προσθέσουμε μεθόδους στους ΑΤΔ μας που επιστρέφουν επαναλήψιμα αντικείμενα, έτσι που να μπορούμε να χρησιμοποιήσουμε την επανάληψη for-each στα περιχόμενά τους

Επαναλήπτες Λίστας στην Java

- Η Java χρησιμοποιεί τον ΑΤΔ ListIterator για τις βασισμένες σε κόμβο λίστες.
- Ο επαναλήπτης αυτός περιλαμβάνει τις παρακάτω μεθόδους:
  - add(e): προσθήκη του e στην παρούσα θέση του δρομέα
  - hasNext(): true αν υπάρχει στοιχείο μετά τον δρομέα
  - hasPrevious: true αν υπάρχει στοιχείο πριν τον δρομέα
  - previous(): επιστρέφει το στοιχείο e πριν το δρομέα και μετακινεί το δρομέα πριν το e
  - next(): επιστρέφει το στοιχείο e μετά τον δρομέα και μετακινεί το δρομέα μετά το e
  - set(e): αντικαθιστά που επέσρεψε η τελευταία next ή previous πράξη με το e
  - remove(): διαγράφει το στοιχείο που επέστρεψε η τελευταία next ή previous μέθοδος

ΑΤΔ ακολουθίας

- Ο ΑΤΔ Sequence είναι η ένωση των ΑΤΔ Πίνακα Λίστας και Λίστας Κόμβων
- Η προσπέλαση στα στοιχεία γίνεται με
  - Δείκτη, ή
  - Θέση
- Πρωταρχικές μέθοδοι :
  - size(), isEmpty()
- Μέθοδοι που βασίζονται σε λίστες πίνακες :
  - get(i), set(i, o), add(i, o), remove(i)
- Μέθοδοι που βασίζονται σε λίστα:
  - first(), last(), prev(p), next(p), replace(p, o), addBefore(p, o), addAfter(p, o), addFirst(o), addLast(o), remove(p)
- Μέθοδοι γεφύρωσης:
  - atIndex(i), indexOf(p)

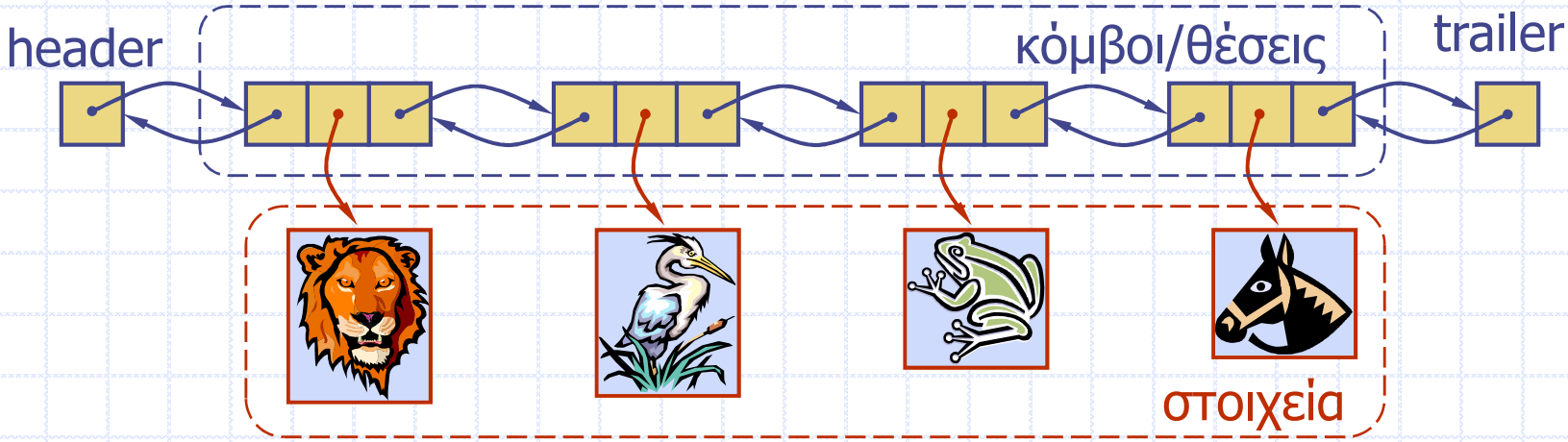
Εφαρμογές των ακολουθιών

- Ο ΑΤΔ ακολουθίας είναι μια βασική, γενικού σκοπού, δομή δεδομένων για αποθήκευση μιας διατεταγμένης σειράς στοιχείων
- Άμεσες Εφαρμογές:
  - Αντικαθιστά στοίβα, ουρά, διάνυσμα, ή λίστα
  - μικρές βάσεις δεδομένων (πχ., βιβλίο διευθύνσεων)
- Έμμεσες εφαρμογές:
  - Ανάπτυξη πιο πολύπλοκων δομών δεδομένων

Υλοποίηση Συνδεδεμένης Λίστας

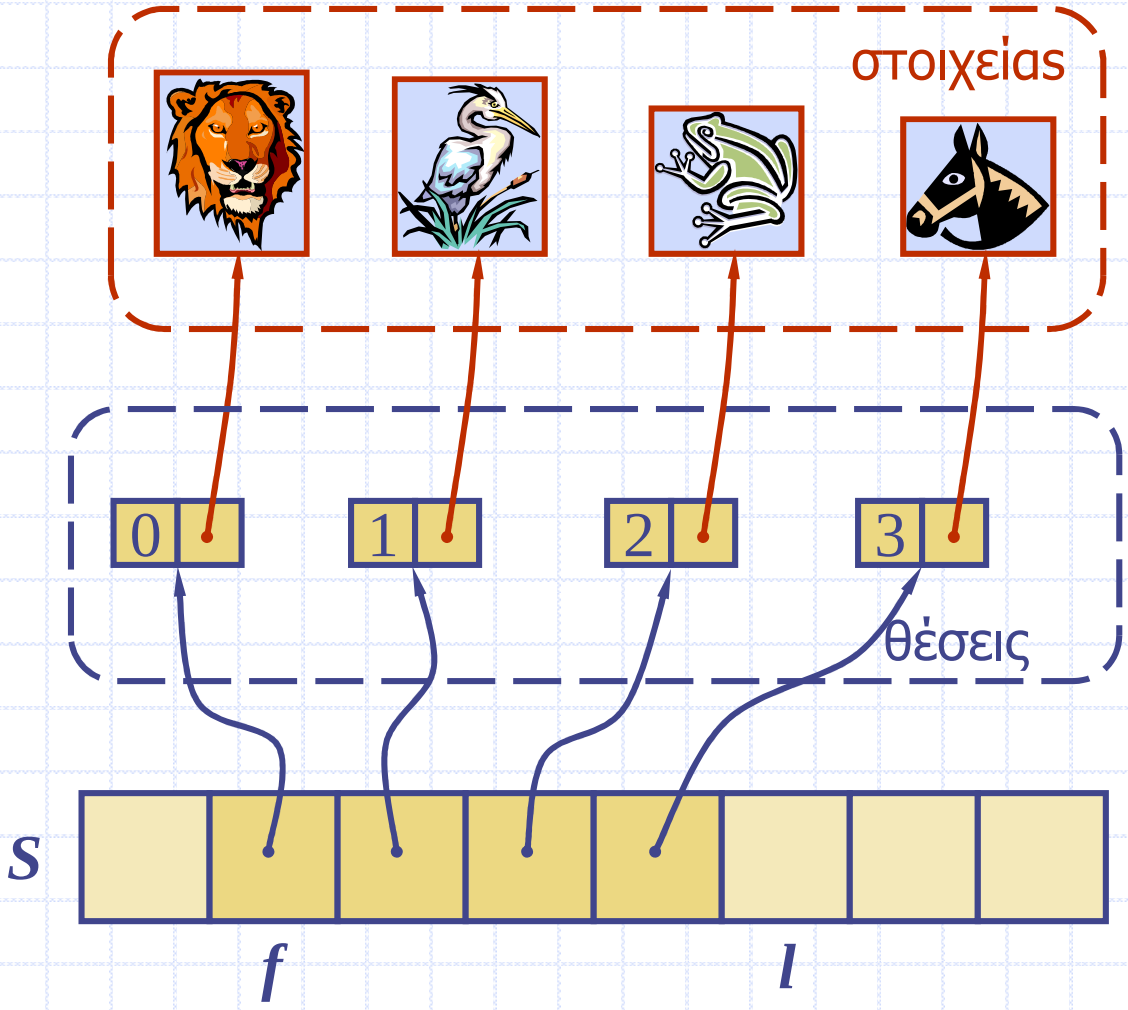
- Μια διπλά συνδεδεμένη λίστα υποστηρίζει μια λογική υλοποίηση του ΑΤΔ ακολουθίας
- Οι κόμβοι υλοποιούν θέση και αποθηκεύουν:
  - στοιχείο
  - σύνδεσμο στον προηγούμενο κόμβο
  - σύνδεσμο στον επόμενο κόμβο
- Ειδικοί κόμβοι αρχής και τέλους

- Οι μέθοδοι που βασίζονται στη θέση τρέχουν σε σταθερό χρόνο
- Οι μέθοδοι που βασίζονται σε δείκτη απαιτούν αναζήτηση από την αρχή προς ή το τέλος ενώ καταγράφουν τους δείκτες• Επομένως τρέχουν σε γραμμικό χρόνο



Υλοποίηση Βασισμένη σε Πίνακες

- Χρησιμοποιούμε ένα κυκλικό πίνακα για αποθήκευση των θέσεων
- Ένα αντικείμενο θέσης αποθηκεύει:
  - Στοιχείο
  - Δείκτη
- Οι δείκτες f και l καταχωρούν την πρώτη και την τελευταία θέση



Σύγκριση Υλοποιήσεων ακολουθιών

Πράξη	Πίνακας	Λίστα
size, isEmpty	1	1
atIndex, indexOf, get	1	n
first, last, prev, next	1	1
set(p,e)	1	1
set(i,e)	1	n
add, remove(i)	n	n
addFirst, addLast	1	1
addAfter, addBefore	n	1
remove(p)	n	1