

# Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής - Εβδομάδα 6

Εβδομάδα 6:  
Συλλογές δεδομένων  
στην Java

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής Αντώνιος Συμβώνης, EMI, Slide 1

### Συλλογές δεδομένων [collections]

- Οι **συλλογές** [collections] (αναφέρονται και ως «υποδοχείς δεδομένων» [containers]) είναι κλάσεις που χρησιμοποιούνται για την αποθήκευση αντικειμένων
- Αποθηκεύουν μία συλλογή από στοιχεία [elements]

Το αντικείμενο τύπου «συλλογής»

Τα στοιχεία

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής Αντώνιος Συμβώνης, EMI, Slide 2

### Διανύσματα και συνδεδεμένες λίστες

- Οι συλλογές μπορεί να υλοποιηθούν με δύο διαφορετικές δομές δεδομένων: τα διανύσματα και τις συνδεδεμένες λίστες/δομές

διάνυσμα

Συνδεδεμένη λίστα

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής Αντώνιος Συμβώνης, EMI, Slide 3

### Διανύσματα [arrays]

- Υποστηρίζονται ως μέρος της Java
- Διαθέσιμα σε σχεδόν όλες τις γλώσσες προγραμματισμού
- Απευθείας αντιστοίχιση με την αρχιτεκτονική του υπολογιστή

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής Αντώνιος Συμβώνης, EMI, Slide 4

### Συνδεδεμένες δομές

- Μπορεί να υλοποιηθούν σε κάθε γλώσσα που υποστηρίζει αναφορές [references] ή δείκτες [pointers] (σχεδόν όλες οι σύγχρονες γλώσσες προγραμματισμού)

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής Αντώνιος Συμβώνης, EMI, Slide 5

### Εισαγωγή σε συνδεδεμένες λίστες

πριν

μετά

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής Αντώνιος Συμβώνης, EMI, Slide 6

# Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής - Εβδομάδα 6

## Διαγραφή από συνδεδεμένες λίστες

πριν

μετά

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής Αντώνιος Συμβώνης, EMI1, Slide 7

## Άλλες συνδεδεμένες δομές

δένδρο

Γράφημα

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής Αντώνιος Συμβώνης, EMI1, Slide 8

## Σύγκριση

- Ποια δομή υπερτερεί; Τα διανύσματα ή οι συνδεδεμένες δομές;
- Θέματα :
  - Αύξηση μεγέθους [growing]
  - Εισαγωγή στοιχείου
  - Διαγραφή στοιχείου
  - Προσπέλαση
  - Απαιτούμενη μνήμη

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής Αντώνιος Συμβώνης, EMI1, Slide 9

## Διανύσματα μεταβλητού μεγέθους

[Growable arrays]

- Τα διανύσματα στην γνήσια μορφή τους δεν είναι πρακτικά γιατί δεν μπορεί να αυξήσουν το μήκος τους όταν αυτό απαιτείται.
- Η λύση: «**διανύσματα μεταβλητού μεγέθους**»
- Ένα διάνυσμα μεταβλητού μεγέθους χρησιμοποιεί μία κλάση για να αποθηκεύει ένα απλό διάνυσμα. Όταν απαιτείται, η κλάση αντικαθιστά το διάνυσμα το οποίο χρησιμοποιεί με άλλο μεγαλύτερο.
- Το διάνυσμα χρησιμοποιείται με έμμεσο τρόπο.

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής Αντώνιος Συμβώνης, EMI1, Slide 10

## Επανακαταχώρηση διανύσματος

Νέο στοιχείο

Δημιουργεί μεγαλύτερο διάνυσμα  
Αντιγράφει σε αυτό όλα τα στοιχεία  
Εισάγει το νέο στοιχείο  
Διαγράφει το παλιό διάνυσμα

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής Αντώνιος Συμβώνης, EMI1, Slide 11

## Χωρητικότητα [capacity]

- Η επανακαταχώρηση είναι μία «ακριβή» λειτουργία
- Το μέγεθος του διανύσματος πρέπει να είναι μεγαλύτερο από το άμεσα αναγκαίο έτσι ώστε η προσθήκη κάθε νέου στοιχείου να μην προκαλεί επανακαταχώρηση του διανύσματος.
- Μέγεθος [size] σε αντιπαράθεση με χωρητικότητα [capacity]
- Μέγεθος: το λογικό μέγεθος των δεδομένων
- Χωρητικότητα: το φυσικό μέγεθος του διανύσματος

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής Αντώνιος Συμβώνης, EMI1, Slide 12

# Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής - Εβδομάδα 6

## Αύξηση μεγέθους

Σύγκριση: Υλοποίηση βασισμένη σε διανύσματα και σε συνδεδεμένες δομές

Πως υλοποιείται η αύξηση μεγέθους;

- Διάνυσμα: πιθανή επανακαταχώρηση
- Συνδεδεμένη δομή: εύκολη

Αποτέλεσμα: 

Διάνυσμα : Συνδεδεμένη δομή
0 : 1

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής

Αντώνιος Συμβώνης, EMI1, Slide 13

## Εισαγωγή στοιχείων

Με ποιο τρόπο εισάγονται τα στοιχεία;

- Διάνυσμα: άλλα στοιχεία (πιθανώς μεγάλος αριθμός) πρέπει να μετατοπισθούν [shifted] (μόνο εάν τα στοιχεία πρέπει να είναι διατεταγμένα)
- Συνδεδεμένη δομή: εύκολη

Αποτέλεσμα: 

Διάνυσμα : Συνδεδεμένη δομή
0 : 2

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής

Αντώνιος Συμβώνης, EMI1, Slide 14

## Διαγραφή στοιχείων

Με ποιο τρόπο διαγράφονται τα στοιχεία;

- Διάνυσμα: άλλα στοιχεία (πιθανώς μεγάλος αριθμός) πρέπει να μετατοπισθούν (μόνο εάν τα στοιχεία πρέπει να είναι διατεταγμένα)
- Συνδεδεμένη δομή: εύκολη

Αποτέλεσμα: 

Διάνυσμα : Συνδεδεμένη δομή
0 : 3

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής

Αντώνιος Συμβώνης, EMI1, Slide 15

## Εισαγωγή στοιχείων

Στην περίπτωση που η διάταξη των στοιχείων δεν είναι απαραίτητη:

- Διάνυσμα: μπορεί να πραγματοποιηθεί με δυο καταχωρήσεις – όμοια με τις συνδεδεμένες δομές

Αποτέλεσμα: 

Διάνυσμα : Συνδεδεμένη δομή
1 : 3

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής

Αντώνιος Συμβώνης, EMI1, Slide 16

## Προσπέλαση στοιχείων

Με ποιο τρόπο προσπελάζονται τα στοιχεία;

- Διάνυσμα: άμεση, μια απλή εντολή, σταθερός χρόνος
- Συνδεδεμένη δομή: διαπέραση της λίστας από την αρχή – πολύ χρονοβόρα διαδικασία στην χειρότερη περίπτωση

Αποτέλεσμα: 

Διάνυσμα : Συνδεδεμένη δομή
2 : 3

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής

Αντώνιος Συμβώνης, EMI1, Slide 17

## Απαιτούμενη μνήμη

Πόση μνήμη απαιτείται;

- Διάνυσμα: Μία αναφορά για κάθε στοιχείο (εάν η χωρητικότητα δεν διαφέρει πολύ από τον αριθμό των στοιχείων)
- Συνδεδεμένη δομή: συνήθως περισσότερη (δεν είναι προφανές από τα διαγράμματα – ισχύει στην πράξη)

Αποτέλεσμα: 

Διάνυσμα : Συνδεδεμένη δομή
3 : 3

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής

Αντώνιος Συμβώνης, EMI1, Slide 18

# Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής - Εβδομάδα 6

## Ανασκόπηση

- Όλες οι υλοποιήσεις των συλλογών δεδομένων βασίζονται σε διανύσματα ή σε συνδεδεμένες δομές (ή σε συνδυασμό τους)
- Η κατανόηση των θεμάτων που αφορούν την υλοποίηση των συλλογών είναι πολύ σημαντική στην εκλογή της κατάλληλης κλάσης-συλλογής της Java (με βάση το συγκεκριμένο σκοπό για τον οποίο πρόκειται να χρησιμοποιηθεί)

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής

Αντώνιος Συμβώνης, EMI1, Slide 19

## Συλλογές δεδομένων στην Java

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής

Αντώνιος Συμβώνης, EMI1, Slide 20

## Το πλαίσιο υλοποίησης συλλογών στην Java

- Στην Java 1.1: μερικές (όχι πολλές) υλοποιήσεις συλλογών δεδομένων (**Vector**, **Hashtable**)
- Στην Java 2 (JDK 1.2): Ένα πλαίσιο **[framework]** υλοποίησης συλλογών δεδομένων
- Πλαίσιο υλοποίησης: Ένα σύνολο από διαπροσωπείες **[interfaces]**, κλάσεις και αλγόριθμους για συγκεκριμένους σκοπούς

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής

Αντώνιος Συμβώνης, EMI1, Slide 21

## Το πλαίσιο υλοποίησης συλλογών στην Java

- **Διαπροσωπείες** (τύποι): Οι διαπροσωπείες της Java περιγράφουν διαφορετικούς λογικούς τύπους συλλογών δεδομένων
- **Υλοποιήσεις**: κλάσεις της Java που υλοποιούν συγκεκριμένους λογικούς τύπους και λαμβάνουν υπ' όψιν τους θέματα επίδοσης **[performance]** (χρόνου και μνήμης)
- **Αλγόριθμοι**: στατικές μέθοδοι, διαθέσιμες για διάφορους τύπους συλλογών δεδομένων

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής

Αντώνιος Συμβώνης, EMI1, Slide 22

## Τα πλεονεκτήματα του πλαισίου υλοποίησης

- Μείωση της απαιτούμενης προγραμματιστικής εργασίας
- Αύξηση της ταχύτητας παραγωγής λογισμικού και της ποιότητάς του
- Μεταφερσιμότητα κώδικα
- Ευκολία εκμάθησης της χρήσης των συλλογών
- Επαναχρησιμοποίηση κώδικα

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής

Αντώνιος Συμβώνης, EMI1, Slide 23

## Διαπροσωπείες

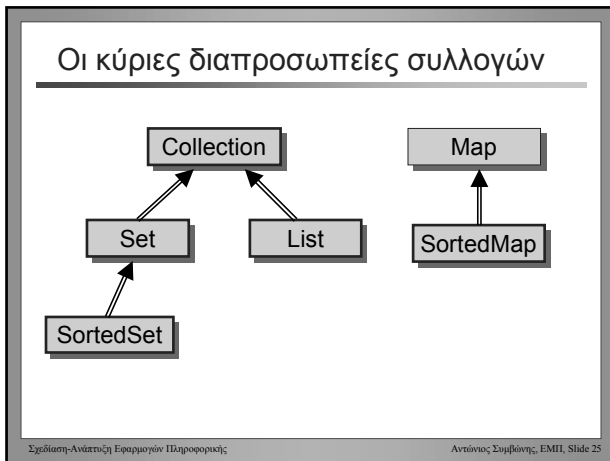
- Η λογική οπτική γωνία των συλλογών δεδομένων
- Επικεντρώνονται στη λειτουργικότητα
- Αφαιρούν/παραβλέπουν τις λεπτομέρειες των υλοποιήσεων

Όταν χρησιμοποιούμε μία συλλογή δεδομένων, πρώτα πρέπει να επιλέξουμε την διαπροσωπεία (τη λειτουργικότητα της συλλογής)

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής

Αντώνιος Συμβώνης, EMI1, Slide 24

# Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής - Εβδομάδα 6



### Οι κύριες διαπροσωπείες συλλογών (2)

**Collection:**  
Ο αφηρημένος υπερτύπος όλων των συλλογών δεδομένων – Ένας υποδοχέας [container] στοιχείων

```
public interface Collection {  
    // Basic Operations  
    int size();  
    boolean isEmpty();  
    boolean contains(Object element);  
    boolean add(Object element);  
    boolean remove(Object element);  
    Iterator iterator();  
  
    // Bulk Operations  
    boolean containsAll(Collection c);  
    boolean addAll(Collection c);  
    boolean removeAll(Collection c);  
    boolean retainAll(Collection c);  
    void clear();  
  
    // Array Operations  
    Object [] toArray();  
    Object [] toArray(Object a[]);  
}
```

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής Αντώνιος Συμβώνης, EMI1, Slide 26

- ### Οι κύριες διαπροσωπείες συλλογών (3)
- **Set**
    - Χωρίς επαναλήψεις στοιχείων, χωρίς διάταξη
    - Παράδειγμα: φοιτητές ενός τμήματος
  - **List**
    - διατεταγμένες (ακολουθίες); Επιτρέπει επαναλήψεις στοιχείων, προσπέλαση με βάση τη θέση του στοιχείου στην ακολουθία [index access], (πχ. Vector)
    - Παράδειγμα: ημερήσιο πρόγραμμα ηλεκτρονικής ατζέντας
  - **Map**
    - Προσπέλαση με βάση τη λέξη-κλειδί [key-value] κάθε στοιχείου (πχ. Hashtable)
    - Ταξινομημένες
- Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής Αντώνιος Συμβώνης, EMI1, Slide 27

- ### Ταξινομημένες συλλογές δεδομένων
- Τα στοιχεία ταξινομούνται αυτόματα
  - Τα στοιχεία κάθε τύπου ταξινομούνται με βάση τον ορισμό του τελεστή ταξινόμησής του
    - Πχ. Οι ακέραιοι ταξινομούνται με βάση την αριθμητική διάταξη, οι ημερομηνίες [Date] με βάση τη χρονολογική διάταξη, οι συμβολοσειρές με βάση την αλφαβητική διάταξη
  - sorted set: παράδειγμα: λίστα φοιτητών τμήματος
  - sorted map: παράδειγμα: λεξικό
- Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής Αντώνιος Συμβώνης, EMI1, Slide 28

### Υλοποίηση ταξινομημένων συλλογών

- Παράδειγμα: συλλογή SortedSet από φοιτητές

Πως λειτουργεί;

- Χρειαζόμαστε έναν ορισμό για την «ταξινόμηση» των αντικειμένων

Λύση: Η διαπροσωπεία Comparable!

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής Αντώνιος Συμβώνης, EMI1, Slide 29

### Η διαπροσωπεία Comparable

- Η διαπροσωπεία Comparable πρέπει να υλοποιείται από τα στοιχεία των ταξινομημένων συλλογών δεδομένων

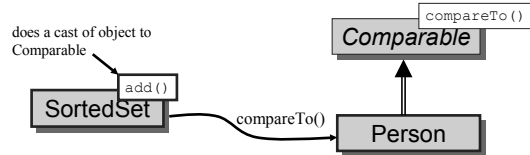
```
public interface Comparable {  
    int compareTo(Object o);  
}
```

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής Αντώνιος Συμβώνης, EMI1, Slide 30

# Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής - Εβδομάδα 6

## Η χρήση της διαπροσωπείας Comparable

- Οι ταξινομημένες συλλογές δεδομένων καλούν τη μέθοδο `compareTo()` των στοιχείων της συλλογής για να επιτύχουν την ταξινόμησή τους



Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής

Αντώνιος Συμβώνης, EMI, Slide 31

## Ανασκόπηση

- Οι κύριες διαπροσωπείες συλλογών :
  - Τρεις βασικοί τύπου συλλογών: Set, List, Map
  - περιγράφουν διαφορετικούς λογικούς τύπους συλλογών δεδομένων
- Έχουμε ήδη μελετήσει:
  - Δυο βασικές τεχνικές υλοποίησης: συνδεδεμένες δομές και διανύσματα

Επομένη ερώτηση : Ποιες είναι οι υλοποιήσεις;  
Ποιες είναι οι κλάσεις;

Σχεδίαση-Ανάπτυξη Εφαρμογών Πληροφορικής

Αντώνιος Συμβώνης, EMI, Slide 32