



ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΠΑΤΡΩΝ  
UNIVERSITY OF PATRAS

ΑΝΟΙΚΤΑ ακαδημαϊκά  
μαθήματα ΠΠ

# Οντοκεντρικός Προγραμματισμός

## Ενότητα 2: Η ΓΛΩΣΣΑ JAVA Σύγκριση JAVA-C

**ΔΙΔΑΣΚΟΝΤΕΣ:** Ιωάννης Χατζηλυγερούδης, Χρήστος  
Μακρής

Πολυτεχνική Σχολή

Τμήμα Μηχανικών Η/Υ & Πληροφορικής

# ΣΥΓΚΡΙΣΗ JAVA - C

# ΤΥΠΟΙ ΔΕΔΟΜΕΝΩΝ (1)

---

## Πρωτογενείς τύποι

- Οι μεταβλητές περιέχουν τις τιμές τους
- Ίδιοι με αυτούς της C (char, short, int, long, float, double)
- Διαφορές:
  - ✓ Υποστηρίζει boolean (true, false), byte (-128, 127). Μεταβλητές τύπου boolean δεν μπορούν να θεωρηθούν σαν ακέραιοι.
  - ✓ Δεν υποστηρίζει μη προσημασμένους αριθμούς
  - ✓ Δεν υποστηρίζει τύπο δείκτη



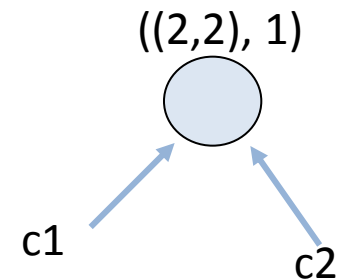
# ΤΥΠΟΙ ΔΕΔΟΜΕΝΩΝ (2)

## Τύποι Αναφοράς

- Οι μεταβλητές είναι μια αναφορά στην τιμή τους (περιέχουν τη διεύθυνσή της)
- αντικείμενα, πίνακες
- Κάθε αναφορά που δεν αναφέρεται σε αντικείμενο ή πίνακα, έχει τιμή null

```
double d;  
Circle c1, c2;  
c1 = new Circle (2.0, 2.0, 1.0);  
c2 = c1;  
c1.r = 4.0;  
d = c2.r;
```

```
double x=10;  
double y = x;  
x = 4.0;
```



# ΑΝΤΙΓΡΑΦΗ-ΣΥΓΚΡΙΣΗ ΑΝΤΙΚΕΙΜΕΝΩΝ

---

## Αντιγραφή

- Δεν γίνεται με ανάθεση/καταχώρηση
- Χρήση της μεθόδου clone (Cloneable Interface)

```
Circle c1 = new Circle();  
c2 = c1.clone();
```

## Σύγκριση

- Δεν γίνεται με έλεγχο ισότητας  
`c1 == c2`
- Δημιουργία μεθόδου σύγκρισης



# ΠΙΝΑΚΕΣ (1)

**Ορισμός** (αντικείμενα που περιέχουν άλλα αντικείμενα)

`int x[];   ή   int [] x;`  1) Δημιουργία αναφοράς

`x= new int[5];`



2) Δημιουργία αντικειμένου και

3) Ανάθεση στην αναφορά

`int x []= new int[5];   ή   int [] x = new int[5];`

`x[0] = 2;`

`x[1] = 3;`



Ανάθεση τιμών στον πίνακα

`int x [] = {2, 3, 5, -6, 9};` (Δημιουργία και αρχικοποίηση)



# ΠΙΝΑΚΕΣ (2)

---

Μέγεθος πίνακα: μεταβλητή `length`  
(η μόνη μεταβλητή της κλάσης πίνακα)

π.χ. `x.length` δίνει το μέγεθος του πίνακα `x`

Η πρόσβαση στα στοιχεία ενός πίνακα γίνεται όπως και στη C.

```
int x [] = new int[10];  
x[0] = 0;  
for (int i=1; i < x.length; i++)  
    x[i] = i + x[i-1];
```



# ΠΙΝΑΚΕΣ (3)

---

## Πολυδιάστατοι πίνακες

```
boolean mat [] [] = new boolean [3] [4];
```

```
int y [] [] [] = new int [3] [4] [5];
```

`mat.length`           (πρώτη διάσταση)

`mat[1].length`       (δεύτερη διάσταση)





# ΑΛΦΑΡΙΘΜΗΤΙΚΑ (1)

**C** → πίνακες χαρακτήρων

**Java** → στιγμιότυπα της κλάσης `String` ή `StringBuffer`  
(δύο ανεξάρτητες κλάσεις)

## Δημιουργία

- Με αυτόματη αναγνώριση-δημιουργία  
π.χ. `String s1 = "Hello";`
- Αποστολή του μηνύματος `new` στην κλάση `String`  
π.χ. `String s2 = new String();`  
`String s3 = new String(s1);`
- Αποστολή του μηνύματος `new` στην κλάση `StringBuffer`  
π.χ. `StringBuffer s4 = new StringBuffer();`  
`StringBuffer s5 = new StringBuffer(s1);`

↓  
Όταν θέλουμε να μπορούμε να τροποποιήσουμε τα αλφαριθμητικά.

(δημιουργοί)

(δημιουργοί)



# ΑΛΦΑΡΙΘΜΗΤΙΚΑ (2)

---

## Βασικές μέθοδοι

- `length()` (επιστρέφει τον αριθμό χαρακτήρων)  
π.χ. `s1.length()`; και όχι ~~`s1.length`~~;
- `charAt(int i)` (επιστρέφει τον χαρακτήρα στη θέση i)  
π.χ. `s1.charAt(1)`; → 'e'

## Πρόσθεση αλφαριθμητικών- τελεστής "+"

`s1 + "John"` → `"Hello John"`



# ΤΕΛΕΣΤΕΣ

---

■ Η Java υποστηρίζει όλους σχεδόν τους τελεστές της C (αριθμητικούς, σύγκρισης, λογικούς) με την ίδια προτεραιότητα.

• Η Java δεν υποστηρίζει

- ✓ τον τελεστή κόμμα ( , ) για συνδυασμό εκφράσεων
- ✓ τους τελεστές δεικτών \* , &
- ✓ τον τελεστή **sizeof**

• Η Java υποστηρίζει επί πλέον

- ✓ τον τελεστή + (και += ) για συνένωση αλφαριθμητικών
- ✓ τον τελεστή **instanceof** (π.χ. s1 instanceof String)
- ✓ τους τελεστές &, | (AND, OR αντίστοιχα σε boolean)  
(πρβλ. &&, ||)



# ΠΡΟΤΑΣΕΙΣ ΕΛΕΓΧΟΥ ΡΟΗΣ

*if-else, while, do-while, for*

- ίδια σύνταξη
- προσοχή στην <έκφραση-συνθήκη> (τύπου boolean, όχι int ή άλλου τύπου)

```
int i = 10 ;
while (i-->0) {
    Circle c1 = new Circle ();
    if (c1 != null) {
        int j;
        do {
            :
        } while (j != 0);
    }
}
```

```
int i = 10 ;
while (i-- > 0) {
    Circle c1 = new Circle ();
    if (c1 != null) {
        int j;
        do {
            :
        } while (j != 0);
    }
}
```



# ΔΗΛΩΣΕΙΣ ΠΑΚΕΤΟΥ

## Ορισμός πακέτου

`package <όνομα πακέτου>;` (πρώτη πρόταση στο αρχείο)

Π.χ. `package transport ;` (προεραϊτικά-ανώνυμο πακέτο)

## Εισαγωγή στοιχείων πακέτου

`import <περιγραφή>;`

Π.χ. `import java.awt.Graphics;  
import java.awt.*;  
import transport.car;`

πακέτο  
κλάση  
μέθοδος

## Προσδιορισμός στοιχείων πακέτων Java

`java.<όνομ-πακέτου>.<όνομ-κλάσης>.<όνομ-μεθόδου>`

Π.χ. `Java.lang.String.substring()`



# ΕΙΣΟΔΟΣ-ΕΞΟΔΟΣ

---

Είσοδος ορισμάτων από γραμμή εντολών

```
java <όνομα-αρχείου> <ορίσματα>
```

Π.χ. `java PrintArgs Kyprou 15 Patra`

```
class PrintArgs {  
    public static void main(String args[]) {  
        for (int i = 0; i < args.length; i++)  
            System.out.println("Argument " + (i+1) + ": " + args[i]);  
    }  
}
```

**Argument 1: Kyprou**

**Argument 2: 15**

**Argument 3: Patra**



# Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στο πλαίσιο του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Αθηνών**» έχει χρηματοδοτήσει μόνο την αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



# Σημείωμα Ιστορικού Εκδόσεων Έργου

---

Το παρόν έργο αποτελεί την έκδοση 1.0.





# Σημείωμα Αναφοράς

---

Copyright: Πανεπιστήμιον Πατρών, Ιωάννης Χατζηλυγερούδης, 2015.  
«Οντοκεντρικός Προγραμματισμός». Έκδοση: 1.0.1 Πάτρα 2015. Διαθέσιμο  
από τη δικτυακή διεύθυνση:

<https://eclass.upatras.gr/courses/CEID1105/>



# Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά, Μη Εμπορική Χρήση Παρόμοια Διανομή 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».



[1] <http://creativecommons.org/licenses/by-nc-sa/4.0/>

Ως **Μη Εμπορική** ορίζεται η χρήση:

- που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο
- που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο
- που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο

Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

# Διατήρηση Σημειωμάτων

---

Οποιαδήποτε αναπαραγωγή ή διασκευή του υλικού θα πρέπει να συμπεριλαμβάνει:

- το Σημείωμα Αναφοράς
- το Σημείωμα Αδειοδότησης
- τη δήλωση Διατήρησης Σημειωμάτων
- το Σημείωμα Χρήσης Έργων Τρίτων (εφόσον υπάρχει)

μαζί με τους συνοδευόμενους υπερσυνδέσμους.



# Σημείωμα Χρήσης Έργων Τρίτων

---

