

Αντικειμενοστρεφής Τεχνολογία

Από τη διάλεξη

«Η Java ως επέκταση της C»

Kleanthis Thramboulidis

Prof. of Software and System Engineering

CEID, University of Patras

<https://sites.google.com/site/thramboulidiskleanthis/>

Η Java ως επέκταση της C

- Τύποι Δεδομένων
- Τελεστές
- Προτάσεις ροής ελέγχου
- Functions vs Methods

Τύποι Δεδομένων

- διαθέτει μικρό σύνολο πρωτογενών τύπων
- υποστηρίζει τη δημιουργία οποιουδήποτε σύνθετου διαμέσου των μηχανισμών
 - της **κλάσης**,
 - του **interface** και
 - του **πίνακα**

Τύποι Δεδομένων/ Πρωτογενείς Τύποι

- **υιοθετεί τους τύπους της C** με τις παρακάτω διαφοροποιήσεις
 - υποστηρίζει επιπλέον τους τύπους **byte** και **boolean**
 - καθορίζει αυστηρά το μέγεθος κάθε τύπου
 - δεν υποστηρίζει απρόσημους αριθμούς
 - Java 8 adds support for unsigned ints and longs
 - δεν υποστηρίζει τον τύπο του δείκτη
 - ορίζει για κάθε τύπο μια default τιμή
 - Η τιμή αυτή αποδίδεται σε κάθε δήλωση μεταβλητής του τύπου, αν ο προγραμματιστής δεν ορίσει αρχική τιμή (*Instance variables have default values. For numbers, the default value is 0, for Booleans it is false, and for object references it is null. Values can be assigned during the declaration or within the constructor.*)
 - local/block variables don't get the default values (should be initialized).

Οι βασικοί τύποι της Java

Τύπος	Περιλαμβάνει	Αρχική τιμή	Μέγεθος	Ελάχιστη Τιμή	Μέγιστη Τιμή
boolean	true or false	false	1 bit	N.A	N.A
char	Unicode character	\u0000	16 bits	\u0000	\uFFFF
byte	signed integer	0	8 bits	-128	127
short	signed integer	0	16 bits	-32768	32767
int	signed integer	0	32 bits	-2147483648	2147483647
long	signed integer	0	64 bits	-9223372036854775808	9223372036854775807
float	IEEE 754 floating-point	0.0	32 bits	$\pm 3.40282347E+38$	$\pm 1.40239846E-45$
double	IEEE 754 floating-point	0.0	64 bits	$\pm 1.79769313486231570E+308$	$\pm 4.94065645841246544E-324$

Ο τύπος Boolean

- οι μεταβλητές τύπου **Boolean** δεν μπορούν
 - να χρησιμοποιηθούν ως ακέραιοι αριθμοί και
 - να μετατραπούν από και προς άλλον τύπο
- έμμεση μετατροπή του ακεραίου *i* σε **boolean** *b*,
b=(i!=0);
- έμμεση μετατροπή του **boolean** *b* σε ακέραιο *i*
i=(b)?1:0;

Δηλώσεις μεταβλητών

- Γενικά ακολουθεί τη σύνταξη της C

<τύπος> <όνομα μεταβλητής>

π.χ. int num;
 char ch;
 float fl;

- Μπορούν να δηλώνονται και μέσα στον κώδικα

```
for(int i = 0 ; i<20 ; i++)
```

Τύποι Δεδομένων / τύποι αναφοράς

- Τα αντικείμενα και οι πίνακες στην Java ονομάζονται και τύποι αναφοράς (**reference types**) γιατί η διαχείριση τους βασίζεται στην έννοια της αναφοράς

- Αντιγραφή αντικειμένου

Δεν γίνεται με ανάθεση καταχώρηση αλλά με την Μέθοδο `clone`

```
Circle c1 = new Circle();
```

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

Object cloning refers to the creation of an exact copy of an object. It creates a new instance of the class of the current object and initializes all its fields with exactly the contents of the corresponding fields of this object.

- Σύγκριση

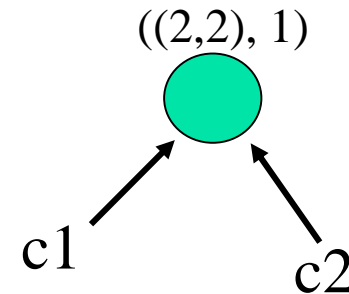
Δεν γίνεται με Έλεγχο ισότητας αλλά με δημιουργία μεθόδου σύγκρισης

```
c1 == c2
```

We can use `==` operators for reference comparison (address comparison) and `.equals()` method for content comparison. In simple words, `==` checks if both objects point to the same memory location whereas `.equals()` evaluates to the comparison of values in the objects.

Τύποι Δεδομένων / τύποι αναφοράς

```
Circle c1, c2;  
c1 = new Circle (2.0, 2.0, 1.0);  
c2 = c1;
```



Τύπος Πίνακα (1)

- δημιουργία πίνακα

int [] x; // δημιουργία αναφοράς

// δημιουργία αντικειμένου και
//ανάθεση του στην αναφορά

x = new int [10];


- αναφορά στο μέγεθος του πίνακα


<όνομα αναφοράς πίνακα>.length

Τύπος Πίνακα(2)

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)

Τύπος Πίνακα (3)

- πρόσβαση στα στοιχεία πίνακα

```
Circle[] circles = new Circle[12];
```

```
for (int i = 0; i < circles.length; i++)  
    circles[i] = new Circle(point[i],  
                             radius[i]);
```

Αλφαριθμητικά

- Θεώρηση της **C**
 - τα αλφαριθμητικά είναι πίνακες χαρακτήρων
- Θεώρηση της **Java**
 - το αλφαριθμητικό είναι αντικείμενο με σαφώς ορισμένη συμπεριφορά που περιγράφεται από την κλάση **String**
 - κάθε αλφαριθμητικό είναι στιγμιότυπο αυτής της κλάσης και συμπεριφέρεται όπως αυτή ορίζει

Τελεστές

- 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. &&, ||)

The **sizeof** operator is the most common operator in C. It is a compile-time unary operator and used to compute the size of its operand. It returns the size of a variable. It can be applied to any data type, float type, pointer type variables.

instanceof is a binary operator used to test if an object is of a given type. The result of the operation is either true or false. It's also known as type comparison operator because it compares the instance with type. ~~Before casting an unknown object, the instanceof check should always be used.~~

- δυαδικός + (string concatenation)

string + string

string + int

string + float

```
String x = "10";
```

```
int y = 20;
```

```
String z = x + y; // z will be 1020 (a String)
```

- **==, !=**
 - επιτρέπουν τη σύγκριση ανάμεσα σε αναφορές
- **λογικοί &&, &, ||, |, !**
 - εκτός απο τον **short-circuit** υπολογισμό (&&,||) προσφέρονται και οι **&, |** τελεστές που υπολογίζουν και τα δύο ορίσματα

We can use == operators for reference comparison (address comparison) and . equals() method for content comparison.

& is a bitwise operator and compares each operand bitwise.

It is a binary AND Operator and copies a bit to the result if it exists in both operands.

&& is a logical AND operator and operates on boolean operands. If both the operands are true, then the condition becomes true otherwise it is false.

& is to be used during bitwise operations and && is useful during logical operations.

■ instanceof

- ελέγχει αν ένα αντικείμενο είναι του δεδομένου τύπου ή υποκλάση του

π.χ.

```
if (someObject instanceof Stack)
    ((Stack)someObject).push(element);
```

Προτάσεις ελέγχου ροής

■ Επανάληψης

- for
- while
- do ... while

■ διακλάδωσης υπό συνθήκη

- if if ... else
- switch (see also **switch expression** in Java 13)

■ διακλάδωσης χωρίς συνθήκη

- break
- continue

γενικά
διατηρείται η
σύνταξη της C

Προτάσεις *if/else*, *while* και *do/while*

- διατηρείται η C σύνταξη
- η έκφραση στην οποία βασίζονται οι υπό συνθήκη προτάσεις πρέπει να είναι τύπου **boolean** και όχι ακέραιου τύπου ή τύπου αναφοράς

```
int i= 10;
while( i- - ) {           // λάθος
    Circle c1= new Circle();
    if ( c1 ) {          // λάθος
        int j;
        do { ...
            } while ( j );    // λάθος
        }
    }
```

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

Προτάσεις `break`, `continue`

`break [label];`

`label_one:`

```
for(int i = 0; i < len ; i++) {
```

```
  if(i < len/2) {
```

```
    while (e1) {
```

```
      if(e2)
```

```
        break label_one;
```

```
      :  
    }
```

```
  :  
}
```

```
  :  
}
```



`continue [label];`

`label_one:`

```
for( i = 0; i < len ; i++) {
```

```
  :
```

```
  while(e1) {
```

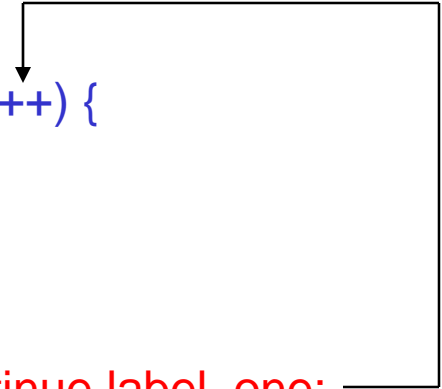
```
    :
```

```
    if(e2)
```

```
      continue label_one;
```

```
    :  
  }
```

```
  :  
}
```



An unlabeled `break` statement terminates the innermost `switch`, `for`, `while`, or `do-while` statement, but a labeled `break` terminates an outer statement.

Σταθερές

- Οι σταθερές έχουν τύπο
 - Δηλώνονται όπως και οι μεταβλητές με προσθήκη δύο προσδιοριστών
 - `static`
 - `final`
- static final** double PI = 3,14...;
- Όπως και στις C συνηθίζεται να γράφονται με κεφαλαία

Αντιμετώπιση απαριθμητικών

```
public enum Day {  
    SUNDAY, MONDAY, TUESDAY, WEDNESDAY,  
    THURSDAY, FRIDAY, SATURDAY }  
}
```

```
public class EnumTest {  
    Day day;  
    main() {  
        ....  
        switch (day) {  
            case MONDAY:  
                ...  
            case TUESDAY :  
                ...  
            ...  
        }  
    }  
}
```

Enumeration (or enum) is a user defined data type in C. It is mainly used to assign names to integral constants, the names make a program easy to read and maintain.

In Java enum is a special "class" that represents a group of constants (unchangeable variables, like final variables).

Functions vs Methods

- A method defines behavior like a function.
However:
 - Methods are defined inside a class definition (establish a relationship between the class and the method)
 - The invocation of a method is different from the call of a function.
- **Java has no functions**
- C++ and Python have both methods and functions

Λοιπές διαφορές από την C (1)

- δυνατότητα δήλωσης τοπικών μεταβλητών σε κάθε μπλοκ κώδικα
- επιτρέπει υπερφόρτωση μεθόδων
- Δεν υποστηρίζει καθολικές μεταβλητές (τις υποστηρίζει όμως έμμεσα)
- δεν υποστηρίζει τη λέξη κλειδί **typedef**
- δεν υποστηρίζει **struct**, **union** και **bitfields**
 - Υποστηρίζει Bitwise and Bit Shift Operators

The C language contains the typedef keyword **to allow users to provide alternative names for the primitive (e.g., int) and user-defined (e.g struct) data types**. Remember, this keyword adds a new name for some existing data type but does not create a new type.

A structure is a key word that create user defined data type in C/C++. A structure creates a data type that can be used to group items of possibly different types into a single type. How to create a structure? 'struct' keyword is used to create a structure.

A union is a user-defined type similar to structs in C except for one key difference. Structures allocate enough space to store all their members, whereas unions can only hold one member value at a time.

Λοιπές διαφορές από την C(2)

- δεν χρησιμοποιεί το void σε περίπτωση που μια μέθοδος δεν δέχεται παραμέτρους
- υποστηρίζει **exception handling**
- υποστηρίζει πολλαπλά νήματα (**multiple threads**)
- Διαθέτει **Garbage Collector**
- υποστηρίζει γραφική διεπαφή (**awt, swing, ...**)