

Il paradigma di programmazione a oggetti

Programmazione
Corso di laurea in Informatica

Paradigma a oggetti

- Formalizza mediante le classi il concetto di modulo che incapsula i dati e le procedure per modificarli
- Le classi si definiscono in una struttura gerarchica e ereditano caratteristiche e funzionalità
- Obiettivo: migliorare l'efficienza del processo di produzione e mantenimento del software

AA 2007/08
© Alberti

2

Programmazione
5. Il paradigma di programmazione a oggetti

Concetti base della programmazione OO

- **Astrazione**
 - Il meccanismo con cui si specificano le caratteristiche peculiari di un oggetto che lo differenziano da altri
- **Incapsulamento dei dati**
 - Il processo con cui si nascondono i dettagli di definizione degli oggetti, solo le interfacce con l'esterno devono essere visibili
- **Ereditarietà**
 - Gli oggetti sono definiti in una gerarchia ed ereditano dall'immediato parente caratteristiche comuni, che possono essere specializzate
- **Polimorfismo**
 - Possibilità di eseguire funzioni specializzate per una particolare classe ma che hanno lo stesso nome

AA 2007/08
© Alberti

3

Programmazione
5. Il paradigma di programmazione a oggetti

Java

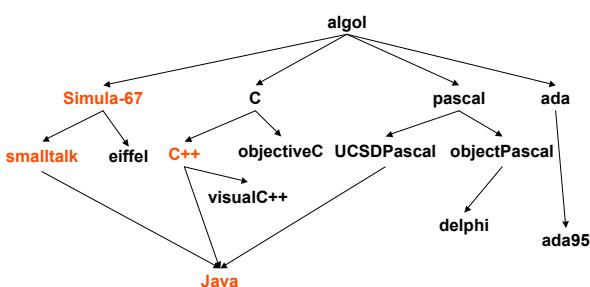
- Java definito dalla Sun Microsystems, Inc.
- Introdotto nel 1995
- E' un linguaggio **orientato agli oggetti**
- Derivato da Smalltalk e C++
- Definito per essere trasportabile su architetture differenti e per essere eseguito da browser

AA 2007/08
© Alberti

4

Programmazione
5. Il paradigma di programmazione a oggetti

Pedigree di Java



AA 2007/08
© Alberti

5

Programmazione
5. Il paradigma di programmazione a oggetti

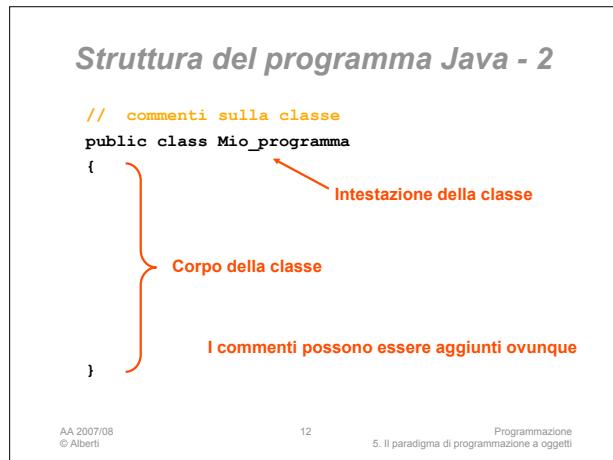
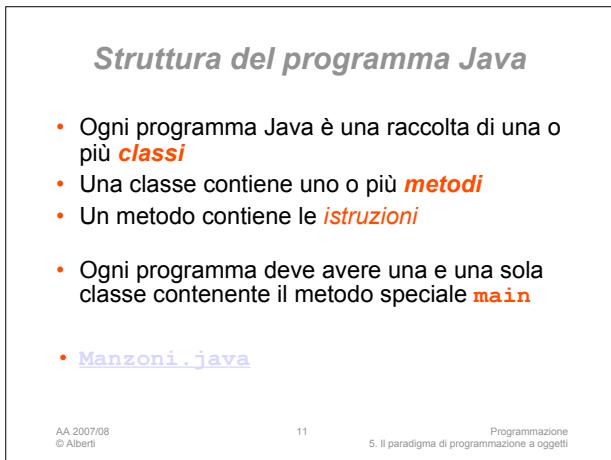
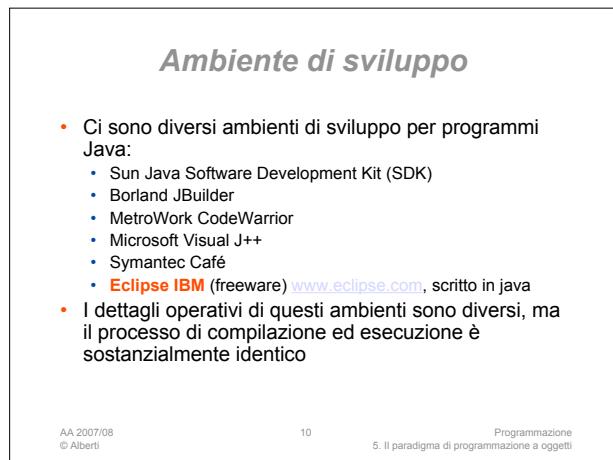
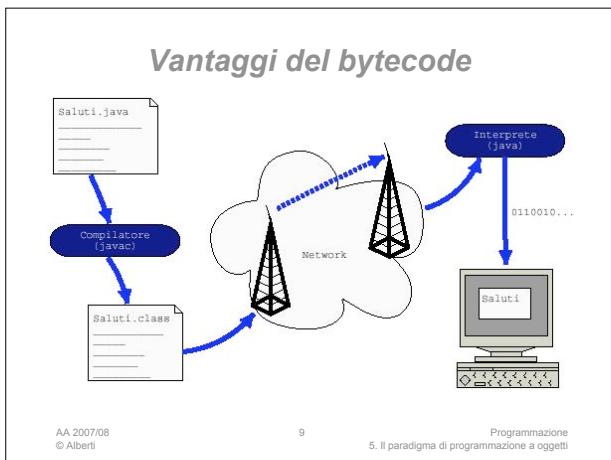
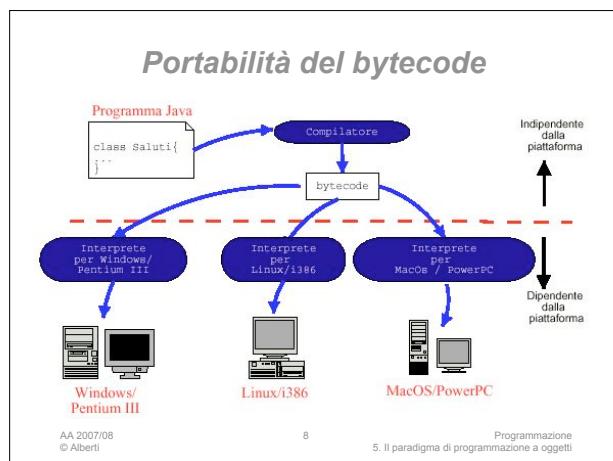
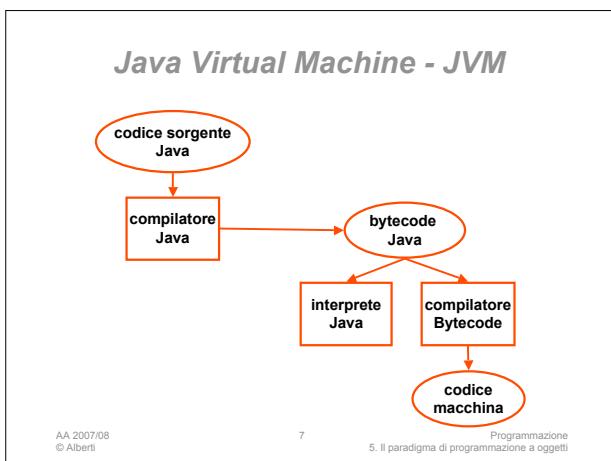
Traduzione e esecuzione di Java

- Il **compilatore** Java traduce il programma **sorgente** in una rappresentazione speciale detta **bytecode**
- Il bytecode Java non è un linguaggio macchina di una CPU particolare, ma di una **macchina virtuale Java**
- Il compilatore Java non è legato ad una particolare macchina
 - Java è **indipendente** dall'architettura della macchina
- L'**interprete** traduce il bytecode nel linguaggio macchina e lo esegue

AA 2007/08
© Alberti

6

Programmazione
5. Il paradigma di programmazione a oggetti



Separatori e operatori

- Separatori: sono caratteri che permettono di separare o raggruppare parti di codice

() [] { } ; , .

Operatori: sono simboli o sequenze di simboli che denotano alcune operazioni

= > < ! ~ ? :

== <= >= != && || ++ --

+ - * / & | % ^ << >> >>>

+= -= *= /= &= |= ^= %= <<= >>= >>>=

AA 2007/08
© Alberti

19

Programmazione
5. Il paradigma di programmazione a oggetti

Costanti

- Una **costante** è un identificatore il cui valore non può essere modificato dopo la sua dichiarazione iniziale
- Il compilatore segnala un errore se si cerca di modificare una costante
- Si dichiara con il modificatore **final**

`final int ALT_MIN = 69;`

- Rendono chiara la semantica dei valori letterali altrimenti non evidenti
- Rendono il codice modulare, facilitandone il cambiamento
- Prevengono errori involontari

AA 2007/08
© Alberti

20

Programmazione
5. Il paradigma di programmazione a oggetti

Riassumendo

- Alcuni identificatori sono definiti da noi (come **Manzoni**)
- Altri sono stati definiti da altri programmati e noi li usiamo (come **println**)
- Alcuni identificatori speciali sono detti **parole riservate** e hanno un significato prestabilito
- Una parola riservata non può essere ridefinita

AA 2007/08
© Alberti

21

Programmazione
5. Il paradigma di programmazione a oggetti

Errori

- Errori di sintassi, che vengono intercettati dal compilatore (**errori di compilazione**)
 - Se c'è un errore durante la fase di compilazione, non viene creato un programma eseguibile
- Errori generati durante l'esecuzione (**errori d'esecuzione**)
 - Tentativi di divisione per zero, che causano la fine anomala del programma
- Errori che producono risultati diversi da quelli desiderati (**errori logici**)

AA 2007/08
© Alberti

22

Programmazione
5. Il paradigma di programmazione a oggetti

Errori sintattici o di compilazione

- Nel programma **Manzoni.java** si sostituisca `System.out.println ("il cielo di ...");` con le seguenti espressioni
- `System.aut.println ("il cielo di ...");`
- `System.out.println ("il cielo di ..._");`
- `System.out.println ("il cilo di ...");`

AA 2007/08
© Alberti

23

Programmazione
5. Il paradigma di programmazione a oggetti

Errori semanticci

- Nel programma **Divisione.java** si provi a passare in input 0 come divisore
- In esecuzione si genera il messaggio:

```
java.lang.ArithmetricException: /by zero
      at Divisione.main(Divisione.java:26)
Exception in thread "main" Process
Exit...
```

AA 2007/08
© Alberti

24

Programmazione
5. Il paradigma di programmazione a oggetti

Errori logici

- Si vuole calcolare il MCD tra due numeri
- Si fornisce il programma [MCD_errato.java](#)
 - Non si ottengono errori in compilazione,
 - né errori in esecuzione,
 - ma non si ottiene neanche il risultato voluto.

Librerie di I/O

- Per esercizi anni scorsi
 - Scaricare il file **cs1.jar**
 - Salvarlo nella cartella **ext** sotto la directory **jdk**
 - **\jdk\jre\lib\ext**
- Per esercizi del testo Pighizzini
 - Scaricare il file **corsoAlberti.jar**
 - E collocarlo nella stessa cartella **ext**