

La Programmazione Object-Oriented in JAVA

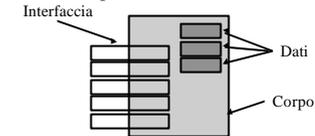
Marco Botta
Dipartimento di Informatica
Universita' di Torino
Corso Svizzera 185
10149 Torino
Tel. 011 6706721 Fax. 011 751603
e-mail: botta@di.unito.it

Programma del corso

- Introduzione alla programmazione Object-Oriented
 - concetti principali
 - terminologia
- Caratteristiche distintive di JAVA
- Sintassi del linguaggio
- Input/Output
- Trattamento delle Eccezioni
- Threads
- Networking
- Applets

Concetti Principali

- Riusabilità
 - la possibilità di riutilizzare un intero sistema software o parti di esso nello sviluppo di altri prodotti software
- Modularità
 - Un modulo consiste di una *interfaccia* e di una *implementazione*
 - Chi usa il modulo conosce solo l'interfaccia
 - L'implementazione è *nascosta*



Concetti Principali (cont.)

- Modularità (cont.)
 - Tre categorie di Moduli
 - Moduli che offrono solo funzioni e procedure e non contengono dati
 - Strutture Dati Astratte (ADS): una ADS consiste di dati e funzioni: i dati sono *nascosti* e possono essere modificati solo dalle funzioni del modulo
 - Tipi di Dati Astratti (ADT): un ADT è simile a una ADS ma rappresenta un tipo di dato

Concetti Principali (cont.)

- Tipo di Dato Astratto
 - un ADT definisce una struttura dati e l'insieme delle operazioni che possono essere eseguite su una particolare istanza dell'ADT
 - I dati e l'implementazione sono *privati* in modo da non poter essere alterati accidentalmente
 - Un ADT funziona esattamente come un tipo di dato di base, quale **int** o **float**.

Concetti Principali (cont.)

- Ereditarietà
 - Meccanismo per costruire nuovi ADT partendo da quelli esistenti *ereditandone* le proprietà
 - Riutilizzo del codice
 - Programmazione per differenza
- Polimorfismo
 - Permette di utilizzare lo stesso *nome* per identificare operazioni simili che differiscono per l'implementazione

Caratteristiche degli oggetti

- **Identità:** i dati sono raggruppati in entità discrete e distinguibili chiamate *oggetti*.
 - paragrafo di un documento
 - finestra su una workstation
 - un pedone degli scacchi

Figura 1.1

- oggetti del mondo reale

Caratteristiche degli oggetti (cont.)

- **Classificazione:** oggetti con la stessa struttura dati (*attributi*) e lo stesso comportamento (*operazioni*) sono raggruppati in una *classe*.
 - Paragrafo
 - Finestra
 - Pezzo degli Scacchi
- Una classe è un'astrazione che descrive proprietà importanti per un'applicazione e ignora il resto
- Una classe descrive un insieme possibilmente infinito di oggetti

Caratteristiche degli oggetti (cont.)

- Ogni oggetto è una *istanza* della sua classe
- Ogni istanza ha i propri valori degli attributi ma condivide nomi degli attributi e operazioni con le altre istanze della classe

Figura 1.2

Caratteristiche degli oggetti (cont.)

- **Polimorfismo:** la stessa operazione può avere comportamenti diversi in classi diverse
 - **move per finestre e move per pezzo degli scacchi**
- Un'operazione è un'azione o trasformazione che un oggetto esegue o subisce
 - **right-justify, display, move**
- Una specifica implementazione di un'operazione si chiama *metodo*.

Caratteristiche degli oggetti (cont.)

- Poiché un operatore è polimorfo può avere più di un metodo che lo implementa
- Nel mondo reale, un'operazione è un'astrazione di comportamenti simili tra tipi di oggetti diversi
- Ogni oggetto *sa come* eseguire le proprie operazioni
- Un linguaggio Object-Oriented sceglie automaticamente il metodo corretto da eseguire in base al nome dell'operazione e alla classe dell'oggetto

Caratteristiche degli oggetti (cont.)

- **Ereditarietà:** condivisione di attributi e operazioni tra classi basata su una relazione gerarchica
- Una classe può essere definita sommariamente e poi raffinata in *sottoclassi*
- Ogni sottoclasse incorpora o *eredita* tutte le proprietà della sua *sopraclasse* e ne aggiunge di proprie
 - **ScrollingWindow, FixedWindow**
- Riduce le ripetizioni e aumenta la riusabilità

Terminologia

- Oggetto: entita` base della programmazione Object-Oriented
 - istanza di un ADT che possiede proprieta` (*variabili di istanza*) e fornisce operazioni (*metodi*) definite per quel particolare tipo di dato
 - un oggetto esegue determinate azioni in risposta a un *messaggio*
 - l'azione eseguita dipende dal *messaggio* e dall'*oggetto* che riceve il messaggio

Terminologia

- *Class*: costruito base per definire tipi di dati astratti (oggetti)
- *instance variables*: descrivono i dati contenuti nel tipo di dato astratto
- *member functions*: definiscono le operazioni possibili sul tipo di dato



Encapsulation

Terminologia

- *extended classes*: ereditano instance variables e member functions dalle classi genitrici e possono essere differenziate aggiungendo nuove variabili e member functions



Inheritance

Terminologia

- *overriding*: permette di ridefinire l'implementazione di member functions ereditate da una classe genitrice



Dynamic Binding

La funzione eseguita e` decisa a tempo di esecuzione

Terminologia

- *Function name overloading*: riduce la necessita` di usare complessi nomi di funzione rendendo il codice di piu` facile lettura



Polymorphism

Funzioni con stesso nome ma numero e tipo dei parametri diversi

Caratteristiche di JAVA

- Semplice
 - facile da imparare conoscendo principi di OOP
 - molto simile a C e C++
 - pochi costrutti da conoscere
- Object-Oriented:
 - un programma JAVA e` costituito dalla definizione di un insieme di oggetti che interagiscono per realizzare la funzionalita` del programma

Caratteristiche di JAVA (cont.)

- Distribuito
 - progettato per sviluppare facilmente applicazioni in rete
- Sicuro
 - l'implementazione del linguaggio garantisce un certo livello di sicurezza e fornisce meccanismi per la protezione del codice
 - difficile se non impossibile creare virus che infettino un sistema

Caratteristiche di JAVA (cont.)

- Robusto
 - pensato per produrre software affidabile e robusto
 - linguaggio fortemente tipato: compile-time e run-time type checking
 - gestione della memoria tramite allocazione e garbage collection automatici
 - sofisticati meccanismi di trattamento delle eccezioni

Caratteristiche di JAVA (cont.)

- Interpretato
 - un programma JAVA è tradotto in un linguaggio intermedio, detto *bytecodes*, che viene interpretato
 - interprete e ambiente run-time costituiscono la JAVA Virtual Machine
- Buone prestazioni
 - circa 20 volte più lento del C
 - sufficiente per applicazioni interattive
 - compilabile per una data CPU

Caratteristiche di JAVA (cont.)

- Indipendente dall'architettura
 - programmi eseguibili su qualsiasi piattaforma che implementa l'interprete e l'ambiente run-time
 - PC, Unix, Mac, etc.
- Portabile
 - programmi compilabili su diverse piattaforme senza necessità di modifica
 - non ci sono aspetti dipendenti dall'implementazione nel linguaggio
 - ambiente portabile

Caratteristiche di JAVA (cont.)

- Multithreaded
 - fornisce il supporto per l'esecuzione di threads multiple
 - migliora le prestazioni interattive
 - primitive di sincronizzazione direttamente incluse nel linguaggio
- Dinamico
 - si adatta all'ambiente
 - le classi possono essere caricate dinamicamente

Caratteristiche di JAVA (cont.)

- Libreria di classi predefinita
 - forniscono astrazioni indipendenti dall'implementazione
 - es.
 - Abstract Windowing Toolkit
 - TCP/IP networking
 - Input/Output
 - Facilmente estendibile con user defined classes