

# Programmazione Orientata agli Oggetti I

Presentazione del Corso  
Anno Accademico 2004-2005

<http://www.db.unibas.it/users/mecca/corsi/progOggettiI>

## Sommario

- Riferimenti
- Il Messaggio Fondamentale
- Modalità Didattiche e Crediti
- Imparare a Programmare a Oggetti
- Metodologia Didattica
- Obiettivi Formativi
- Materiale Didattico
- Eccezioni



## Riferimenti

- Sito del Corso  
www.db.unibas.it/users/mecca/corsi/progOggetti
- Giansalvatore Mecca  
mecca@unibas.it  
http://www.db.unibas.it/users/mecca
- Ricevimento studenti
  - ⇒ lunedì 18:30-19:30
  - ⇒ mercoledì 12:00-13:00



## Il Messaggio Fondamentale

*Il Corso di Programmazione a Oggetti I  
è un corso centrale del nostro piano di studi  
(uno dei più importanti tra quelli del nucleo)*

*E' il primo di una serie di insegnamenti coordinati  
sulla programmazione a Oggetti, che seguirà con  
Programmazione a Oggetti II e Tecnologie Web*

*Rispetto al corso di Programmazione Procedurale  
gli obiettivi didattici sono molto più elevati:  
(a) insegnare le metodologie della POO  
(b) insegnare a sviluppare software di qualità  
industriale utilizzando le piattaforme selezionate*



## Modalità Didattiche e Crediti

- Corso di 6 crediti
  - ⇒ 6 CFU: approx. a 150 ore totali di lavoro
  - ⇒ lezioni frontali: 5 crediti: approx. 50 ore
  - ⇒ laboratorio: 1 credito: approx. 15 ore
- Totale: circa 65 ore di didattica assistita
  - ⇒ il resto è **studio**
- Propedeuticità
  - ⇒ Programmazione Procedurale



## Modalità Didattiche e Crediti

- Impostazione della didattica assistita
  - ⇒ in aula vengono impartiti i concetti
  - ⇒ e vengono mostrati molti esempi di codice
  - ⇒ a casa lo studente studia il materiale delle lezioni
  - ⇒ in laboratorio vengono proposti esercizi da risolvere con l'assistenza del docente, ma non vengono proposti ulteriori concetti
  - ⇒ ulteriori esercizi da risolvere a casa



## Modalità Didattiche e Crediti

### ○ Esercitazioni di laboratorio

⇒ martedì ore 14:00-19:00

⇒ due turni: 14:00 – 16:30 e 16:30 – 19:00

⇒ a cominciare dal 3 maggio (approx.)

⇒ per partecipare alle esercitazioni è necessario iscriversi al corso utilizzando il servizio di iscrizione in linea sul portale (le iscrizioni saranno aperte successivamente)

### ○ Assistenza alle esercitazioni

⇒ Dott. Salvatore Raunich



## Modalità Didattiche e Crediti

### ○ La frequenza

⇒ non è obbligatoria ma fortemente consigliata

### ○ Calendario della didattica

⇒ 4 aprile – 13 maggio: lezioni

⇒ 16 maggio – 27 maggio: prove intercorso

⇒ 31 maggio – 8 luglio: lezioni

⇒ 11 luglio – 31 luglio: prove finali



## Modalità Didattiche e Crediti

- Modalità di svolgimento delle prove finali
  - ⇒ prova scritta (questionari)
  - ⇒ prova pratica di programmazione (livello base, intermedio, avanzato)
  - ⇒ eventuale prova orale
- Prova intercorso
  - ⇒ prova scritta (questionario)
  - ⇒ consente di scegliere il livello della prova finale



## Modalità Didattiche e Crediti

- Date delle prove
  - ⇒ prova intercorso: 20 maggio
  - ⇒ I appello: 12 luglio
  - ⇒ II appello: 25 luglio
  - ⇒ III appello: 30 settembre
  - ⇒ IV appello: dicembre 2005
  - ⇒ V appello: febbraio 2006



## Imparare a Programmare a Oggetti

### ○ Ipotesi

- ⇒ supponiamo che lo studente conosca almeno un linguaggio programmazione procedurale
- ⇒ come in tutti i corsi, anche in questo corso lo studente deve imparare varie cose

### ○ In particolare

- ⇒ la tecnologia dei linguaggi a oggetti
- ⇒ le piattaforme applicative a oggetti
- ⇒ i metodi della programmazione a oggetti



## Imparare a Programmare a Oggetti

### ○ La tecnologia dei linguaggi a oggetti

- ⇒ cosa sono le classi, cosa sono gli oggetti
- ⇒ come si scrivono le classi
- ⇒ come si manipolano gli oggetti

### ○ I metodi della programmazione a oggetti

- ⇒ come si scelgono le classi
- ⇒ come si progetta e si sviluppa un'applicazione a oggetti



## Imparare a Programmare a Oggetti

- Le piattaforme applicative
  - ⇒ gli strumenti (compilatore, debugger ecc.)
  - ⇒ la libreria di classi predefinite
- In particolare
  - ⇒ l'ultimo punto è fondamentale
  - ⇒ es: per conoscere bene il linguaggio Java bisogna conoscerne la libreria di classi predefinita
  - ⇒ varie migliaia di classi e di metodi (>>)



## Metodologia Didattica

- Caratteristiche del corso
  - ⇒ questo corso è didatticamente molto complesso ("rischioso")
  - ⇒ prevede una continuazione nel corso di Programmazione a Oggetti II (coordinamento)
  - ⇒ la didattica non sarà organizzata a strati



## Metodologia Didattica

- Programmazione a Oggetti I
  - ⇒ centrato sulle tecnologie a oggetti
  - ⇒ metodi della programmazione basata su oggetti
- Programmazione a Oggetti II
  - ⇒ centrato sulle metodologie a oggetti
  - ⇒ metodi della programmazione orientata agli oggetti



## Metodologia Didattica

- La programmazione basata su oggetti
  - ⇒ tecniche fondamentali per definire le classi
  - ⇒ per creare e manipolare gli oggetti
- In particolare
  - ⇒ la programmazione con i componenti
  - ⇒ l'incapsulamento
  - ⇒ gli strati applicativi
  - ⇒ la sintassi e la semantica dei linguaggi a oggetti





## Metodologia Didattica

### ○ Obiettivo

- ⇒ *“programmare le stesse cose che si possono programmare in C, ma in modo più semplice e con qualità più alta”*
- ⇒ consentire di sviluppare soluzioni a problemi di piccola/media dimensione, producendo applicazioni migliori di quelle sviluppabili con la programmazione procedurale
- ⇒ senza dover utilizzare gli aspetti avanzati della programmazione orientata agli oggetti



## Metodologia Didattica

### ○ La programmazione orientata su oggetti

- ⇒ introduzione agli aspetti caratterizzanti della programmazione nei linguaggi a oggetti

### ○ In particolare

- ⇒ ereditarietà e polimorfismo
- ⇒ classi interne
- ⇒ e applicazioni (introduzione ai design pattern, framework, programmazione grafica)



## Metodologia Didattica

### ○ Obiettivo

- ⇒ *“programmare cose che in pratica in C non si possono programmare”*
- ⇒ sviluppare soluzioni a problemi di dimensioni medio/grandi
- ⇒ utilizzando soluzioni sperimentate (framework e design pattern)
- ⇒ e gestendo la variabilità delle specifiche



## Metodologia Didattica

### ○ Attenzione

- ⇒ i due livelli non sono completamente separati
- ⇒ per apprendere la programmazione basata sugli oggetti bisogna avere nozioni sull'ereditarietà
- ⇒ queste nozioni verranno introdotte superficialmente nel primo corso
- ⇒ e poi approfondite nel secondo



## Il Nucleo: Una Vista d'Insieme

-elementi di base  
-tecniche algoritmiche di base  
-fondamenti di programmazione

Prog. Procedurale

-componenti (classi e oggetti)  
-incapsulamento  
-strati applicativi

Prog. Oggetti I  
P. Basata su Oggetti

-ereditarietà, polimorfismo  
-classi interne  
-framework  
-programmazione grafica

Prog. Oggetti II  
P. Orientata agli Oggetti

Reti

Basi di  
Dati

-programmazione  
client/server  
-programmazione  
Web

Tecnologie di Sviluppo  
per il Web

Ingegneria del  
Software



## Metodologia Didattica

### ○ Alcune tecniche didattiche

- ⇒ doppia tecnologia
- ⇒ presentazione degli argomenti centrata sul codice
- ⇒ metafora consistente (>>)

### ○ Doppia tecnologia

- ⇒ due linguaggi di riferimento: Java e C#
- ⇒ due piattaforme: Java 2 SE e .NET



## Metodologia Didattica

- Presentazione degli argomenti
  - ⇒ centrate sul codice
  - ⇒ uno o più progetti per ciascun argomento
- Struttura tipica di un argomento
  - ⇒ I: introduzione con esempi
  - ⇒ II: presentazione dei concetti
  - ⇒ III: conclusioni con i particolari
  - ⇒ IV: secondo linguaggio



## Obiettivi Formativi

- Livello di base
  - ⇒ capacità di risolvere problemi di programmazione standard con i componenti
  - ⇒ capacità di selezionare ed organizzare i componenti (architettura applicativa corretta)
  - ⇒ conoscenza della piattaforma Java
  - ⇒ conoscenza del linguaggio UML



## Obiettivi Formativi

### ○ Livello intermedio

- ⇒ prevede tutte le competenze previste dal livello base e in aggiunta:
- ⇒ capacità di risolvere problemi di programmazione di media complessità con i componenti
- ⇒ conoscenza della piattaforma .NET



## Obiettivi Formativi

### ○ Livello avanzato

- ⇒ prevede tutte le competenze previste dal livello intermedio e in aggiunta:
- ⇒ capacità di risolvere problemi di programmazione complessi con i componenti
- ⇒ capacità di costruire modelli a oggetti corretti per problemi di grandi dimensioni
- ⇒ conoscenza di tecniche di programmazione (test di regressione, logging, riflessione)



## Materiale Didattico

### ○ Composizione del materiale

- ⇒ materiale delle lezioni (in linea sul sito)
- ⇒ vari libri consigliati
- ⇒ collegamenti ad altro materiale in linea

### ○ Nota

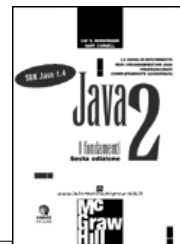
- ⇒ il materiale delle lezioni è già disponibile sul sito
- ⇒ potrebbero esserci aggiustamenti locali



## Materiale Didattico

### ○ Libri consigliati su Java

- ⇒ Horstmann, Cornell – Java 2, i Fondamenti (6a Edizione, 2003) – *McGraw Hill Libri Italia - Milano* - 50 Euro circa
- ⇒ un buon manuale sul linguaggio Java
- ⇒ copre la maggior parte degli argomenti del corso
- ⇒ l'impostazione è tradizionale

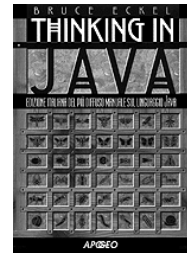




## Materiale Didattico

### ○ Libri consigliati su Java (continua)

- ⇒ Eckel – Thinking in Java
- ⇒ disponibile in linea in lingua inglese sul sito [www.bruceeckel.com](http://www.bruceeckel.com)
- ⇒ edizione italiana edita da Apogeo
- ⇒ un libro classico sul linguaggio
- ⇒ ha pregi e difetti rispetto al primo



## Materiale Didattico

### ○ Libri su C#

- ⇒ non è indispensabile acquistarne nessuno
- ⇒ sono disponibili gratuitamente vari tutorial in linea sulla rete
- ⇒ e molti capitoli del libro C# Essentials
- ⇒ in alternativa, qualsiasi manuale è adeguato per gli scopi di questo corso



## Eccezioni

- Studenti degli anni successivi al II
  - ⇒ per tutto l'anno accademico 2004-2005 è possibile optare tra nuovo e vecchio programma
- Nuovo programma
  - ⇒ il programma e le prove di questo corso
- Vecchio programma
  - ⇒ programma e prove del corso "edizione speciale" 2003-2004



## Eccezioni

- Edizione Speciale 2003-2004
  - ⇒ vari argomenti in più (ereditarietà e polimorfismo, librerie, es: date, design pattern, programmazione grafica);
  - ⇒ alcuni argomenti in meno (test di regressione, logging)
  - ⇒ modalità diversa di svolgimento delle prove





## Eccezioni

- Modalità di svolgimento delle prove
  - ⇒ una prova pratica uguale per tutti
  - ⇒ una prova orale
  - ⇒ le prove si svolgeranno in ambiente Windows
  - ⇒ saranno disponibili le trasparenze delle lezioni



## Riassumendo

- Riferimenti
- Il Messaggio Fondamentale
- Modalità Didattiche e Crediti
- Imparare a Programmare a Oggetti
- Metodologia Didattica
- Obiettivi Formativi
- Materiale Didattico
- Eccezioni