

Programmazione e laboratorio: introduzione al corso

Stefano Mizzaro

Dipartimento di matematica e informatica
Università di Udine
<http://www.dimi.uniud.it/mizzaro>
mizzaro@dimi.uniud.it

Programmazione per TWM, Lezione 1
27 settembre 2004

<http://www.dimi.uniud.it/mizzaro/dida/Prog0405/>

Oggi

- Presentazione del corso
 - I docenti
 - L'esame
 - Raccomandazioni
 - Programma preliminare del corso e testi
- Inizio
 - I primi esempi di programmi Java
 - Come si scrive un programma

Stefano Mizzaro - Introduzione

2

Docenti: teoria

- Stefano Mizzaro: (24 ore, prime 12 lezioni)
 - <http://www.dimi.uniud.it/mizzaro>
 - mizzaro@dimi.uniud.it
- Paolo Coppola: (24)
 - <http://www.dimi.uniud.it/coppola>
 - coppola@dimi.uniud.it

Stefano Mizzaro - Introduzione

3

Docenti: laboratorio

- Lucio Ieronutti: lab. (12)
 - <http://www.dimi.uniud.it/ieronutt/>
 - ieronutt@dimi.uniud.it
- Stefano Burigat: lab. (12)
 - <http://www.dimi.uniud.it/burigat/>
 - burigat@dimi.uniud.it
- Marco Frailis: lab. (24)
 - <http://??>
 - frailis@fisica.uniud.it

Stefano Mizzaro - Introduzione

4

Il corso

- 48 ore di teoria, 6 CFU:
 - Mizzaro (24), Coppola (24)
- 48 ore di laboratorio, 4 CFU:
 - Ieronutti (12), Burigat (12), Frailis (24)
- Orario lezioni
- Divisione in due per il laboratorio!!
 - Lab. A: A – Matt.
 - Lab. B: Mila. – Z
 - Fiscali (almeno all'inizio)
 - Niente lab. questa settimana!

Stefano Mizzaro - Introduzione

5

Orario

	Lun	Mar	Mer	Gio	Ven
I 9:00-10:45		Lab A (A – Matt)	Prog (E)		
II 11:00-12:45	Prog (E)	Lab B (Mila – Z)	Lab A (A – Matt)		
"II e ½"					
III 15:00-16:45			Lab B (Mila – Z)		
IV					

Stefano Mizzaro - Introduzione

6

Modalità d'esame preliminari (1/3)

- Scritto + (progetto +) orale
- Scritto
 - [2 Provette: dic. e mar. (fine I e II per. did.)]
 - Niente appunti, libri, ... allo scritto/provetta
 - 2a provetta in contemporanea/alternativa al primo appello
 - Se voto scritto $\leq 10 \Rightarrow -5$ all'appello succ.!!

Stefano Mizzaro - Introduzione

7

Modalità d'esame preliminari (2/3)

- Progetto (opzionale)
 - Gruppi di 3 persone (eccezioni da concordare preventivamente, ad es.: lavoro)
 - Il progetto va consegnato allo scritto e spedito per email (dettagli insieme al progetto)
 - Pubblicato sul web ca. 2 settimane prima dell'appello
 - Voto max. senza progetto: 27
 - Progetto dà incremento di 0-3 punti
 - Solo se voto scritto ≥ 21 !!
 - Se voto < 21 , non consegnato, non presentato \Rightarrow progetto annullato (come non fatto)

Stefano Mizzaro - Introduzione

8

Modalità d'esame preliminari (3/3)

- Orale
 - Su tutto il programma (e sul progetto)
 - Va sostenuto nello stesso appello del progetto, altrimenti bisogna rifare il progetto
 - Tutti i componenti del gruppo devono sapere TUTTO sul progetto come se l'avessero fatto da soli
 - Chi dimostra di non sapere abbastanza deve rifare scritto e progetto (chi non ha passato lo scritto...)

Stefano Mizzaro - Introduzione

9

Cose importanti (1/2)

- Studiate!
- (198 ... 55, 142... 86, 130... 63; tot.: 470... 204)
- Non basta studiare: usate il calcolatore!
 - Seguite le lezioni di laboratorio
 - Andate in laboratorio per conto vostro
 - Se avete un calcolatore...
- Fate domande! (anche per email)
- Venite a ricevimento!
 - Orario: Sindy (Mar. 10:00 – 12:00)

Stefano Mizzaro - Introduzione

10

Cose importanti (2/2)

- Bast-Che-Funz: chi se ne frega di come è scritto un programma, basta che funzioni...
- SE LA PENSATE COSI', NON PASSATE L'ESAME!
- DOVETE imparare a scrivere programmi di buona qualità
- Anche chi sa già programmare imparerà qualcosa...

Stefano Mizzaro - Introduzione

11

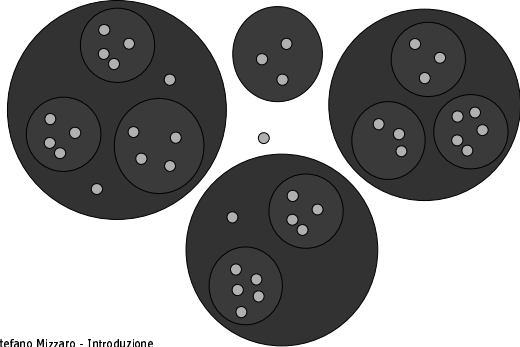
Programma preliminare del corso: obiettivi

- Introdurre gli aspetti fondamentali
 - della programmazione strutturata,
 - dell'occultamento delle informazioni,
 - della programmazione orientata agli oggetti
- Imparare a programmare
 - Usiamo il linguaggio Java
 - Java è il mezzo, non il fine!
- Accennare a computabilità e complessità

Stefano Mizzaro - Introduzione

12

Analisi sistematica: dai mattoni più piccoli alle componenti più grandi



Stefano Mizzaro - Introduzione

13

Programma preliminare

1. La programmazione strutturata (22h)
2. Tipi di dati astratti e occultamento delle informazioni (4h)
3. I fondamenti della programmazione orientata agli oggetti (10h)
4. Le API del Java (10h)
5. Cenni alle teorie della computabilità e complessità (2h)

Stefano Mizzaro - Introduzione

14

1. La programmazione strutturata (22h)

- L'ABC della programmazione:
 - Espressioni
 - Strutture di controllo
 - Sviluppo incrementale
 - Array
 - Metodi
 - Ricorsione
- Presenti in ogni linguaggio di programmazione

Stefano Mizzaro - Introduzione

15

2. Tipi di dati astratti, occultamento delle informazioni (4h)

- Come fare programmi più grandi e complicati...
- ...e comunque modificabili, estendibili, comprensibili, ...
- Tipi di dati astratti, occultamento delle informazioni
- Concetti e realizzazione in Java
- Verso l'OO ("Object Oriented")

Stefano Mizzaro - Introduzione

16

3. I fondamenti della programmazione OO (10h)

- Evoluzione dai TDA: oggetti, classi, istanze, scambio messaggi
- Eredità
- Polimorfismo
- La programmazione OO in Java
 - Classi astratte
 - Interfacce
 - Package

Stefano Mizzaro - Introduzione

17

4. Le API del Java (10h)

- Application Programming Interface
- "Librerie"
- Il programmatore di oggi deve conoscere anche le API
- Introduzione, cenni:
 - File, stringhe, utilità, ...
 - Interfacce utente grafiche
 - Applet

Stefano Mizzaro - Introduzione

18

5. Teorie della computabilità e della complessità (cenni, 2h)

- Teoria matematiche (teoremi!)
- Computabilità
 - Cos'è un programma?
 - "Cosa può fare un programma"?
 - È possibile scrivere un programma che ...?
 - Ci sono linguaggi di programmazione più "potenti" di altri? Cosa vuol dire "più potente"?
- Complessità
 - Dato un problema, ci sono programmi più efficienti di altri per risolverlo? (più veloci, che usano meno risorse, ...)

Stefano Mizzaro - Introduzione

19

Testi e materiale

- S. Mizzaro, Introduzione alla programmazione con il linguaggio Java, Franco Angeli, Milano, 3a ed., 2001, ISBN 88-464-1696-1, <http://www.dimi.uniud.it/mizzaro/LibroJava>
- P. Coppola, S. Mizzaro, Laboratorio di programmazione in Java, Apogeo, Milano, 2004, ISBN: 88-503-2145-7
- Materiale su Web. Sito Web del corso: <http://www.dimi.uniud.it/mizzaro/dida/Prog0405/>
- Dispensa: V. Della Mea, S. Mizzaro. L'algoritmo, ...
- ... altro...

Stefano Mizzaro - Introduzione

20

Collegamenti con altri corsi

- | | |
|--|--|
| <ul style="list-style-type: none"> ■ 1o anno <ul style="list-style-type: none"> ■ Architetture ■ Tecnologie Web ■ 2o anno <ul style="list-style-type: none"> ■ Programmazione OO ■ Algoritmi e strutture dati ■ Sistemi operativi ■ Sistemi multimediali | <ul style="list-style-type: none"> ■ 3o anno <ul style="list-style-type: none"> ■ Ingegneria del software ■ Lab. Tecnologie Lato Server ■ Interazione uomo macchina ■ Progetto siti Web ■ ... |
|--|--|

Stefano Mizzaro - Introduzione

21

Riassunto

- Il corso
 - Docenti
 - Lezioni
 - Esame
 - Raccomandazioni
 - Programma preliminare del corso e testi
- Domande? Osservazioni?
- I primi esempi di programmi Java
- Come si scrive un programma

Stefano Mizzaro - Introduzione

22

Il nostro primo programma in Java

```
/* Il nostro primo programma in Java */
class CiaoATutti {
    public static void main (String[] args) {
        System.out.println("Ciao a tutti!");
    }
}
```

- Prima riga: commento
- Due righe successive: intestazione (saremo più precisi), deve esserci sempre
 - "Nome del programma" (saremo più precisi)
- 4a riga: istruzione di "stampa"
- Poi le graffe chiuse, una per ogni graffa aperta

Stefano Mizzaro - Introduzione

23

L'attività del programmatore

- Ciclo editing-compilazione-esecuzione
- Editing
 - Usando un editor di testo, scrivere il programma e salvarlo su file
- Compilazione
 - Usando un compilatore, tradurre il programma da Java a un linguaggio più vicino all'HW
- Esecuzione
 - Eseguire il programma

Stefano Mizzaro - Introduzione

24

Editing-compilazione-esecuzione

- Editing
 - XEmacs, Blocco Note, ... (formato solo testo!!)
- Compilazione
 - Compilatore Java
 - Non produce codice direttamente eseguibile dall'HW (non produce linguaggio macchina)
 - Produce un "codice a byte" (bytecode)
 - `>javac NomeFile.java`
- Esecuzione
 - Interprete Java
 - Sa eseguire il bytecode
 - `>java NomeProgramma`

Stefano Mizzaro - Introduzione

25

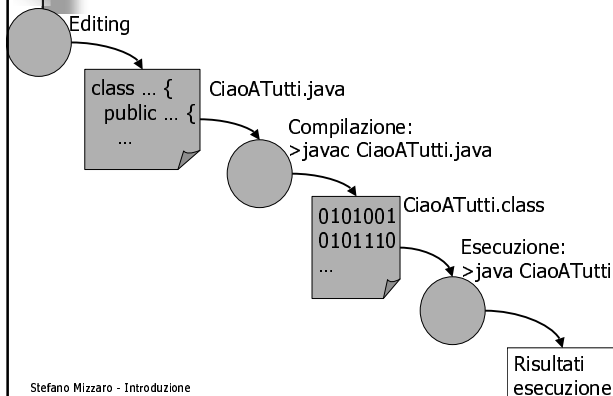
Il JDK

- JDK/SDK (Java/Standard Development Kit)
- La "cassetta degli attrezzi" del programmatore Java
- Compilatore: **javac**
- Interprete: **java**
 - JVM (Java Virtual Machine, Macchina virtuale Java)
 - JRE (Java Runtime Environment) = JVM + ...

Stefano Mizzaro - Introduzione

26

Schema



Stefano Mizzaro - Introduzione

Risultati
esecuzione

Editing-compilazione-esecuzione in pratica

- Scrivo il programma **CiaoATutti** (uso l'editor XEmacs)
- Lo compilo
- Lo eseguo

```

>javac CiaoATutti.java
>java CiaoATutti
Ciao a tutti!
>
  
```

Stefano Mizzaro - Introduzione

28

Un programma più "serio"

```

/* Programma per calcolare l'area di un
triangolo */
class AreaTriangolo {
public static void main (String[] args) {
int base;
int altezza;
int area;

base = 5;
altezza = 12;
area = base * altezza / 2;
System.out.println(area);
}
}
  
```

Stefano Mizzaro - Introduzione

29

Notare

- Parte "dichiarativa" e parte "eseguibile"
- Sequenza di istruzioni (;)
- Assegnamento (=)
- Variabili: dichiarazione e uso
- Espressioni
- **System.out.println**
- Eseguiamo in pratica anche questo
 - E poi proviamolo su altri dati...

Stefano Mizzaro - Introduzione

30

La "leggibilità" (comprensibilità)

```
/* Programma per calcolare l'area di
un triangolo */
class AreaTriangoloDisordinata { public
    static void main
    (
String[] args) { int
base; int altezza; int area; base =
5; altezza = 12; area = base * altezza / 2;
    System.out.println(area);}}
```

- È lo stesso programma? Fa le stesse cose?
- Morale: la leggibilità è importante!

Stefano Mizzaro - Introduzione

31

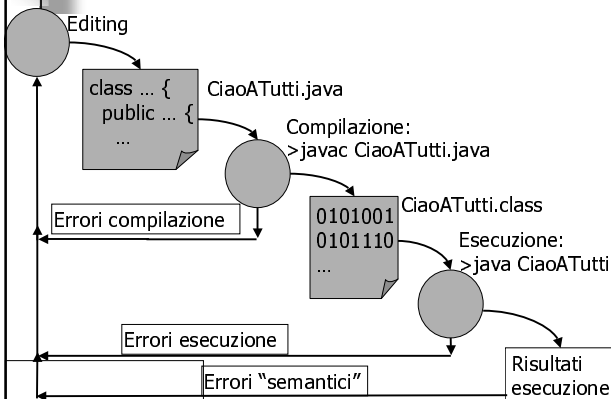
Errori

- Si fanno **sempre** errori quando si scrive un programma!
- Bisogna imparare a gestirli:
 - Capire i messaggi di errore del compilatore
 - Capire i messaggi di errore al tempo di esecuzione
 - Capire perché il programma non fa quello che vogliamo
- "Ciclo" editing-compilazione-esecuzione

Stefano Mizzaro - Introduzione

32

Schema



Un altro programma

```
/* Qual e' il massimo fra 2 cifre? */
import java.io.*;
class Max2 {
    public static void main(String[] args)
        throws IOException {
        int a, b;
        a = System.in.read();
        b = System.in.read();
        if (a>b) {
            System.out.println(
                "Il massimo e' il primo");
        }
        else {
            System.out.println(
                "Il massimo e' il secondo");
        }
    }
}
```

Stefano Mizzaro - Introduzione

34

Osserviamo

- `System.in.read()`
 - throws `IOException`
 - `import java.io.*`
- `System.in.read()`
 - Legge un carattere (!)
- `if ... else ...`
 - Alternative
 - Dopo l'`if` ci va un'espressione logica (condizione)
 - "else" significa "altrimenti"

Stefano Mizzaro - Introduzione

35

Stampare i numeri da 1 a 10 (brutto!!)

```
class Stampa1_10 {
    public static void main(String[] args) {
        System.out.println(1);
        System.out.println(2);
        System.out.println(3);
        System.out.println(4);
        System.out.println(5);
        System.out.println(6);
        System.out.println(7);
        System.out.println(8);
        System.out.println(9);
        System.out.println(10);
    }
}
```

Stefano Mizzaro - Introduzione

36

Stampare i numeri da 1 a 10 (meglio)

```
class Stampa10 {  
    public static void main(String[] args) {  
        int contatore;  
        contatore = 1;  
        while (contatore <= 10) {  
            System.out.println(contatore);  
            contatore = contatore + 1;  
        }  
    }  
}
```

- Questo è molto meglio. Perché:
 - Altrimenti non passate l'esame... ☺
 - ...stampare i numeri da 1 a 1000...

Stefano Mizzaro - Introduzione

37

Fine!

- Prima parte
 - Il corso, i docenti, l'esame
 - Programma preliminare del corso e testi
- Seconda parte
 - I primi esempi di programmi Java
 - Editing-compilazione-esecuzione
- Prossima lezione
 - Altri esempi di programmi
 - Analisi più sistematica
 - "Dai mattoni più piccoli alle componenti più grandi"

Stefano Mizzaro - Introduzione

38