

I metodi - II

Stefano Mizzaro

Dipartimento di matematica e informatica
Università di Udine
<http://www.dimi.uniud.it/mizzaro/>
mizzaro@dimi.uniud.it
Programmazione, lezione 10
29 ottobre 2007

Comunicazioni di servizio

- Domani martedì 30/11 non c'è lezione
- Martedì 6/11 iniziamo alle 14:45
 - (Ultima lezione con me)
- Argomenti ultime 3 lezioni:
 - Metodi (oggi)
 - Metodi + Ricorsione (lunedì)
 - Cenni di computabilità + Esempi (martedì)
 - [Valutazione corso]
- Qs. mercoledì: ricevimento annullato

Stefano Mizzaro - Metodi II

2

Dove siamo

- Mattoni, Programmazione strutturata, Sviluppo incrementale, Array
- Intro metodi/sottoprogrammi
 - Definizione (intestazione e corpo) e uso
 - Parametri formali e attuali, associazione
 - Passaggio parametri per valore
 - Funzioni e procedure
 - Utilità dei metodi
- (Dimenticato: convenzioni Java per nomi di metodi = nomi di variabili)

Stefano Mizzaro - Metodi II

3

Struttura programma Java

```
class ... {
  static <tipo> <id> (<parametri>) {
    ...
  }
  static <tipo> <id> (<parametri>) {
    ...
  }
  public static void main (String[] args) {
    ...
  }
  static <tipo> <id> (<parametri>) {
    ...
  }
  static <tipo> <id> (<parametri>) {
    ...
  }
}
```

Oggi

- Visibilità
- Durata
- Gestione dei metodi nella JVM
- Metodi sovraccarichi
- Funzioni matematiche predefinite
- Esempi

Stefano Mizzaro - Metodi II

5

Visibilità (*scope*)

```
for (int i = 1; i <= 10; i++) {
  ...
}
```

- **i** visibile solo nel corpo del for

```
class Prova {
  public static void main (String[] args) {
    int j = 6;
    for (int i = 1; i <= 5; i++) {
      System.out.println(i);
      System.out.println(j);
    }
    System.out.println(i);
    System.out.println(j);
  }
}
```

```
>javac Prova.java
Prova.java:8: Undefined variable: i
  System.out.println(i);
                   ^
```

1 error

Stefano Mizzaro - Metodi II

6

Blocchi annidati

- Variabili usate in un blocco ma definite in un blocco esterno: no problem
 - (Il blocco interno è dentro al blocco esterno!)
- Variabili con stesso nome in blocchi annidati
- 2 casi:
 1. Locale al metodo e di classe
 - Quella locale *nasconde* quella di classe
 2. All'interno di un blocco e locale al metodo
 - Errore, conflitto di nomi

Stefano Mizzaro - Metodi II 13

Esempio caso 1

```
class Nascoste {
    static int x,y; //Def. var. globali

    static void f() {
        int x;
        x = 1; // Locale
        y = 1; // Globale
        System.out.println(x);
        System.out.println(y);
    }

    public static void main (String[] args) {
        x = 0; // Globale
        y = 0; // Globale
        f();
        System.out.println(x);
        System.out.println(y);
    }
}
```

```
>java Nascoste
1
1
0
1
```

Stefano Mizzaro - Metodi II 14

Esempio caso 2

```
class VariabiliBlocchi {
    public static void main (String[] args) {
        int x = 0;

        if (x > 0) {
            int x = 1;
            x++;
        }
    }
}
```

```
>javac VariabiliBlocchi.java
VariabiliBlocchi.java:6: Variable 'x' is already defined
in this method.
    int x = 1;
    ^
1 error
```

Stefano Mizzaro - Metodi II 15

Variabili "globali" vs. passaggio parametri

- Usate parametri, non variabili globali
- È più semplice capire cosa fa un metodo:
 - basta leggere il metodo e
 - non bisogna far riferimento a tutto il resto del programma
 - (var. globali; altri metodi)

Stefano Mizzaro - Metodi II 16

Esempio

```
class Parametri {
    static int x;

    static void inc() {
        x++;
    }

    public static void main (String[] args) {
        x = 0;
        inc();
        System.out.println(x);
    }
}
```

- **Brutto!**

Stefano Mizzaro - Metodi II 17

Esercizio: traccia di esecuzione

```
class EsercizioScope1 {
    static int x, y;

    static int metodo1(int x) {
        x = 1;
        y = 1;
        System.out.println("metodo1 " + x + " " + y);
        return x + y;
    }

    static void metodo2() {
        int y;
        x = 2;
        y = metodo1(x);
        System.out.println("metodo2 " + x + " " + y);
    }

    public static void main (String[] args) {
        x = 0;
        y = 0;
        metodo2();
        System.out.println("main " + x + " " + y);
    }
}
```

**LA TRACCIA DI ESECUZIONE
È CONDIZIONE NECESSARIA
PER PASSARE L'ESAME!**

ergo

**SE NON SAPETE FARE LA
TRACCIA DI ESECUZIONE
NON PASSATE L'ESAME!**

Stefano Mizzaro - Metodi II 19

Scaletta

- Visibilità
- Durata
- Gestione dei metodi nella JVM
- Metodi sovraccarichi
- Funzioni matematiche predefinite
- Esempi

Stefano Mizzaro - Metodi II 20

La durata (*lifetime*)

- Da quando la memoria viene allocata a quando viene "cancellata"
- Diverso da visibilità:
 - Ad un dato momento dell'esecuzione, una variabile può esistere e non essere visibile
 - (se una variabile non esiste, non può essere visibile...)

Stefano Mizzaro - Metodi II 21

Esempio

```
class Automatiche {
    static void f() {
        int i;
        g();
        ...;
    }
    static void g() {
        ...
    }
    public static void main (String[] args) {
        f();
    }
}
```

- Durante l'esecuzione di **g**, **i** esiste ma non è visibile

Stefano Mizzaro - Metodi II 22

Gestione metodi nella JVM

- Come funziona la JVM
- Per ogni metodo c'è un **record di attivazione**
 - Zona di memoria che contiene le variabili e i parametri formali del metodo
- **Pila (stack) di record di attivazione**
 - Per ogni invocazione di metodo viene allocato un record di attivazione
 - I record di attivazione sono gestiti a Pila (uno sopra l'altro), con politica Last In First Out (LIFO)
 - Il record di attivazione in cima alla pila corrisponde al metodo in esecuzione in quel momento

Stefano Mizzaro - Metodi II 23

Esempio

```
class EsempioPila {
    static void f (int x) {
        int y = 3;
    }
    static void g (int z) {
        int i = 2;
        f(z);
    }
    public static void main (String[] args) {
        int w = 1;
        g(w);
    }
}
```

- Simuliamo l'esecuzione costruendo la pila dei record di attivazione

Stefano Mizzaro - Metodi II 24

Visibilità e durata

- La visibilità è un concetto statico
 - Si guarda il codice, ci si aiuta con il modello a contorni
- La durata è un concetto dinamico
 - Bisogna eseguire il programma
- P.S. Variabili di classe:
 - In un record di attivazione in fondo alla pila
 - Allocato a inizio esecuzione
 - Durata = tutta l'esecuzione

Stefano Mizzaro - Metodi II 25

Esercizio

```
class ... {
    static void f() {
        int x;
        g();
        ...;
    }

    static void g() {
        ...;
    }

    public static void main (String[] args) {
        ...;
        f();
        g();
    }
}
```

- La variabile **x** esiste durante l'esecuzione di **g**?

Stefano Mizzaro - Metodi II 26

Scaletta

- Visibilità
- Durata
- Gestione dei metodi nella JVM
- Metodi sovraccarichi
- Funzioni matematiche predefinite
- Esempi

Stefano Mizzaro - Metodi II 27

Metodi sovraccarichi

- Firma di un metodo =
 - nome + elenco dei (tipi dei) parametri
 - Attenzione: non il tipo restituito!
- "Sovraccarico" (*overload*):
 - Metodi con nome uguale e firma diversa
 - Il nome del metodo è carico di più significati
 - Il Java distingue i metodi sulla base della loro firma

Stefano Mizzaro - Metodi II 28

Esempio

```
class Overloading {
    static int somma (int x, int y) {
        return x + y;
    }

    static int somma (int x, int y, int z) {
        return x + y + z;
    }

    public static void main (String[] args) {
        System.out.println(somma(1,1));
        System.out.println(somma(1,2,3));
    }
}
```

```
static double somma (int x, int y) {
    return (double) x + y;
}
```

Stefano Mizzaro - Metodi II 29

Funzioni matematiche predefinite

- `sqrt(double)`
- `exp(double)`
- `log(double)`
- `cos(double)`
- `sin(double)`
- `tan(double)`
- `pow(double, double)`
- `abs(double)`
- `floor(double)`
- `ceil(double)`
- `min(double, double)`
- `max(double, double)`
- `PI = 3.141592...`
- `E = 2.71828459...`

- Premettere `Math.`

Stefano Mizzaro - Metodi II 30

Esercizi

- Metodo che calcola il MCD fra 2 numeri
- Programma che visualizza i numeri primi fra 2 e n , con n letto in input durante l'esecuzione
- Metodo che visualizza i numeri primi fra 2 e n , con n parametro del metodo
- Metodo che visualizza i primi n numeri primi

Riassunto

- Metodi
 - Visibilità (contorni, regola di visibilità, variabili locali a un metodo, di classe, locali a un blocco)
 - Durata, gestione dei metodi nella JVM, pila dei record di attivazione
 - Metodi sovraccarichi
 - Funzioni matematiche predefinite
 - Esempi
- Libro: quasi tutto il cap. 5.
- Eserciziario: quasi tutto il § 3.1
- Prossima lezione: ancora metodi e ricorsione