

Introduzione alla programmazione strutturata

Stefano Mizzaro

Dipartimento di matematica e informatica
Università di Udine
<http://www.dimi.uniud.it/mizzaro>
mizzaro@dimi.uniud.it
Programmazione, lezione 4
5 ottobre 2005

Esperimento

- Valutazione di un motore di ricerca con clustering
 - CREDO → CREDINO
- Email con
 - Corso di laurea, Anno di corso
 - Orari/disponibilità
 - Esperienza uso palmari/PDA
 - Ad Andrea Della Pietra <andrea.dellapietra@gmail.com>
- Mezz'ora di tempo...
 - Sistema interessante
 - Lavoro di tesi di laurea
 - Esperimento di valutazione
 - ...

Stefano Mizzaro - Prog. strutt. 2

Riassunto

- Esempi
- I mattoni di un programma Java:
 - Parole riservate
 - Identificatori
 - Variabili
 - Letterali
 - Costanti
 - Operatori
 - Espressioni
 - Assegnamento, dichiarazioni di variabile
 - Commenti
 - Tipo
- Puntualizzazioni ed esempi
 - Costanti
 - Valori booleani o logici
 - Operatore condizionale ? :
- Conversione di tipo
- Valori temporanei delle espressioni

Stefano Mizzaro - Prog. strutt. 3

Oggi

- Come mettere insieme i mattoni:
 - Introduzione alle strutture di controllo della programmazione strutturata
 - Sequenza
 - Selezione
 - Iterazione

Stefano Mizzaro - Prog. strutt. 4

Introduzione alla programmazione strutturata

- Un programma è fatto da istruzioni
- Abbiamo visto
 - pezzi di istruzioni
 - (poche) istruzioni complete
 - Dichiarazioni di variabile (con nome e tipo)
 - Assegnamento
 - `System.out.println()`
 - `System.in.read`
 - ...
- Vediamo come metterli insieme

Stefano Mizzaro - Prog. strutt. 5

Le strutture di controllo della programmazione strutturata

- Sequenza
 - ; e {}
- Selezione
 - `if`, `if/else`, `switch/case`
- Iterazione (ripetizione)
 - `while`, `do/while`, `for`
 - (`break`, `continue`)

Stefano Mizzaro - Prog. strutt. 6

Sequenza

- Ogni istruzione "atomica" termina con un punto e virgola ;
- Si possono raggruppare più istruzioni "per farle diventare come un'unica istruzione" con le parentesi graffe { }
- Uno spazio = più spazi = tabulazione = ritorno carrello (a capo)
- Incolonnare per leggibilità
 - Aggiungere spazi (2, 4) ad ogni {
 - Togliere spazi ad ogni }
 - { a fine riga
 - } su riga isolata (qualche eccezione...)

Stefano Mizzaro - Prog. strutt.

7

Esempio

```
/* 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 - Prog. strutt.

8

Esempio

```
class Stampa1_10 {
    public static void main(String[] args) {
        int contatore;
        contatore = 1;
        while (contatore <= 10) {
            System.out.println(contatore);
            contatore = contatore + 1;
        }
    }
}
```

Stefano Mizzaro - Prog. strutt.

9

Selezione binaria

- **if, if/else**
- Alternativa: fai o questo o quello
- Se una certa condizione è vera fai questo altrimenti fai quello
 - "Condizione": espressione booleana (di tipo `boolean`, quindi con valore `true` o `false`)
 - "Questo" e "quello": istruzioni, eventualmente composte

Stefano Mizzaro - Prog. strutt.

10

Esempio

```
/* 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 - Prog. strutt.

Eccezione...

11

Forma generale

- **if** e **else** sono parole riservate
- **c** è una condizione (espr. booleana)
- **I1** e **I2** sono istruzioni
 - Eventualmente composte (fra { })
- Le graffe sono indispensabili solo per istruzioni composte
 - Ma metterle sempre può essere una buona idea...
- La parte **else** può anche non esserci
- Incolonnamento e graffe
 - Rientrare il ramo **if** e il ramo **else**
 - Incolonnare l'**else** sotto l'**if** corrispondente

```
if (C)
    I1
else
    I2
```

```
if (C) {
    I1;
    I2;
} else {
    I3;
    I4;
}
```

```
if (C)
    I1
```

Stefano Mizzaro - Prog. strutt.

12

Diagrammi di flusso

- (flow chart)
- Possono essere utili per capire
- Non per scrivere programmi!
- "Ramo if" e "ramo else"

Stefano Mizzaro - Prog. strutt. 13

Esempi/esercizi

- Scrivere:
 - Un frammento di programma che dice (visualizza in output) se un numero x è positivo o negativo
 - Un frammento di programma che dice se un numero x è positivo o negativo e incrementa di uno il valore assoluto di x
 - Un frammento di programma che dice se un numero x è pari o dispari
 - Un frammento di programma che dice se un numero x è zero, strettamente positivo o strettamente negativo

Stefano Mizzaro - Prog. strutt. 14

Gli if annidati

- if dentro a if (dentro a if...)
- Nel ramo if o nel ramo else
 - Annidamenti if e annidamenti else
- Fare attenzione
- Regola: l'else si riferisce all'ultimo if
- Se la regola non ci dà quello che vogliamo, usiamo le { }

Stefano Mizzaro - Prog. strutt. 15

Esempio

```

if (x >= 0)
  if (x < 10)
    System.out.println("positivo e minore di 10");

```

```

if (x >= 0)
  if (x < 10)
    System.out.println("positivo e minore di 10");
  else
    System.out.println("positivo e maggiore= di 10");

```

```

if (x >= 0)
  if (x < 10)
    System.out.println("positivo e minore di 10");
  else
    System.out.println("negativo");

```

The last code block is crossed out with a red X.

Stefano Mizzaro - Prog. strutt. 16

Preferire gli annidamenti else

- Non raggiungono il margine destro:
 - } else if tutto su una riga

```

if (C1)
  if (C2) {
    if (C3)
      I1;
    } else
      I2;
else
  I3;

```

→

```

if (!C1)
  I3;
else if (!C2)
  I2;
else if (C3)
  I1;

```

Stefano Mizzaro - Prog. strutt. 17

n-aria: switch/case

- Selezione non fra 2 alternative, ma fra più alternative

```

switch (E) {
  case V1: I1; break;
  case V2: I2; break;
  ...
  case Vn: In; break;
  default: I; break;
}

```

Stefano Mizzaro - Prog. strutt. 18

Esempio

```
switch (c) {
  case 'a':
    System.out.println("Vocale");
    break;
  case 'e':
    System.out.println("Vocale");
    break;
  case 'i':
    System.out.println("Vocale");
    break;
  case 'o':
    System.out.println("Vocale");
    break;
  case 'u':
    System.out.println("Vocale");
    break;
  default: System.out.println("Consonante");
    break;
}
```

Stefano Mizzaro - Prog. strutt. 19

Esempio (più sintetico)

```
switch (c) {
  case 'a': // Vai al prox.
  case 'e': // Vai al prox.
  case 'i': // Vai al prox.
  case 'o': // Vai al prox.
  case 'u':
    System.out.println("Vocale");
    break;
  default: System.out.println("Consonante");
    break;
}
```

Stefano Mizzaro - Prog. strutt. 20

Più e meno generale dell'if

- Più generale perché selezione n-aria, non solo binaria
- Meno generale perché la condizione deve essere del tipo `E==Vi`
- (mentre nell'if può essere qualsiasi)
 - Esempio: `x > 0?`
- Vi devono essere espressioni valutabili dal compilatore
 - (un letterale, ...)
- di tipo `byte`, `short`, `int`, `long` o `char`

Stefano Mizzaro - Prog. strutt. 21

Altro esempio (brutto!!)

```
switch (giorno) {
  case 1:
    System.out.println("Lunedì");
    break;
  case 2:
    System.out.println("Martedì");
    break;
  case 3:
    System.out.println("Mercoledì");
    break;
  case 4:
    System.out.println("Giovedì");
    break;
  case 5:
    System.out.println("Venerdì");
    break;
  case 6:
    System.out.println("Sabato");
    break;
  case 7:
    System.out.println("Domenica");
    break;
  default:
    System.out.println("Valore non lecito");
    break;
}
```

Stefano Mizzaro - Prog. strutt. 22

Iterazione

- Istruzioni ripetute
- Ripeti "questo" mentre la condizione è vera
 - "questo", al solito, è un'istruzione, che può anche essere composta
- while (C) I
 - Ripeti l'istruzione I mentre la condizione c è vera
 - Se I è composta, bisogna usare le graffe

Stefano Mizzaro - Prog. strutt. 23

Esempio

```
class CicloWhile {
  public static void main (String[] args) {
    int i;

    i = 1;
    while (i <= 10) {
      System.out.println(i);
      i = i + 1;
    }
    System.out.println("Fine stampa");
  }
}
```

condizione (espressione booleana) istruzione (composta)

Stefano Mizzaro - Prog. strutt. 24

Diagramma di flusso

- Differenze fra **while** e **if**

if (C)
I1

while (C)
I1

Stefano Mizzaro - Prog. strutt. 25

Esempio

- Stampare solo i numeri pari da 1 a 10

```

i = 1;
while (i <= 10) {
    if (i % 2 == 0)
        System.out.println(i);
    i = i + 1;
}
    
```

- Oppure
- (peggio!): codice ripetuto

```

i = 1;
while (i <= 10) {
    if (i % 2 != 0)
        i = i + 1;
    else {
        System.out.println(i);
        i = i + 1;
    }
}
    
```

Stefano Mizzaro - Prog. strutt. 26

Esempio

- Stampare solo i numeri pari da 1 a 10, ancora meglio:
 - Più sintetico
 - Più efficiente

```

i = 2;
while (i <= 10) {
    System.out.println(i);
    i = i + 2;
}
    
```

Stefano Mizzaro - Prog. strutt. 27

Punto della situazione

- Fine "mattoni"
- Strutture di controllo della programmazione strutturata
- Sequenza
 - ; e {}
- Selezione
 - if, if/else, switch/case
- Iterazione (ripetizione)
 - while, do/while, for
 - (break, continue)
- Con mooolta calma...

Stefano Mizzaro - Prog. strutt. 28

Prossima lezione

- Paolo Coppola
- Continuate con la programmazione strutturata
- Come si scrive un programma
- Esercizio: scrivere un programma che trova il maggiore fra quattro caratteri letti in input, lo assegna a una variabile e visualizza "il maggiore e' il primo/secondo/..."

Stefano Mizzaro - Prog. strutt. 29