

## Introduzione alla programmazione strutturata

Stefano Mizzaro

Dipartimento di matematica e informatica  
 Università di Udine  
<http://www.dimi.uniud.it/mizzaro/>  
 mizzaro@uniud.it  
 Programmazione, lezione 4  
 3 marzo 2015

## Riassunto

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

S. Mizzaro - Programmazione strutturata 1

2

## Oggi

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

S. Mizzaro - Programmazione strutturata 1

3

## 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

S. Mizzaro - Programmazione strutturata 1

4

## Le strutture di controllo della programmazione strutturata

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

S. Mizzaro - Programmazione strutturata 1

5

## Sequenza

- Ogni istruzione "atomica" termina con un punto e virgola `;`
- Si possono raggruppare più istruzioni "per farle diventare come un'unica istruzione" (istruzione **composta**) 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...)

S. Mizzaro - Programmazione strutturata 1

6

### 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);
  }
}
    
```

S. Mizzaro - Programmazione strutturata 1 7

### Esempio

```

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

S. Mizzaro - Programmazione strutturata 1 8

### 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

S. Mizzaro - Programmazione strutturata 1 9

### 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");
    }
  }
}
    
```

Eccezione... 10

### 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
    
```

S. Mizzaro - Programmazione strutturata 1 11

### Diagrammi di flusso

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

S. Mizzaro - Programmazione strutturata 1 12

## 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

S. Mizzaro - Programmazione strutturata 1

13

## 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 `{ }`

S. Mizzaro - Programmazione strutturata 1

14

## 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");
```


S. Mizzaro - Programmazione strutturata 1

15

## 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;
```

- N.B. Nego la condizione + scambio rami `if` e `else` => ottengo istruzione equivalente

S. Mizzaro - Programmazione strutturata 1

16

## n-aria: `switch/case`

- Selezione non fra 2 alternative, ma fra più alternative
- Forma (quasi) generale:

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

S. Mizzaro - Programmazione strutturata 1

17

## 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");
}
```

S. Mizzaro - Programmazione strutturata 1

18

## 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;
}
```

S. Mizzaro - Programmazione strutturata 1

19

## 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:  $x > 0$ ?
- Inoltre, le  $Vi$  devono essere espressioni valutabili dal compilatore
  - (un letterale, ...)
- e di tipo `byte`, `short`, `int`, `long` o `char`

S. Mizzaro - Programmazione strutturata 1

20

## 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;
}
```

S. Mizzaro - Programmazione strutturata 1

21

## Le strutture di controllo della programmazione strutturata

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

S. Mizzaro - Programmazione strutturata 1

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
  - I: Corpo del ciclo

S. Mizzaro - Programmazione strutturata 1

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)

S. Mizzaro - Programmazione strutturata 1

24

## Schema tipico di un ciclo

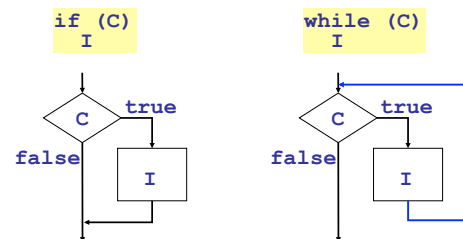
- Condizione su una variabile
    - (inizializzata fuori dal ciclo!)
  - Operazione/i che sfrutta quella variabile
  - Incremento/decremento/modifica/... della variabile
- Ma non è l'unica possibilità!!

S. Mizzaro - Programmazione strutturata 1

25

## Diagramma di flusso

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



S. Mizzaro - Programmazione strutturata 1

26

## 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;
  }
}
```

S. Mizzaro - Programmazione strutturata 1

27

## Esempio

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

- Più sintetico
- Più elegante
- (Più efficiente)

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

S. Mizzaro - Programmazione strutturata 1

28

## 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 molta calma...

S. Mizzaro - Programmazione strutturata 1

29

## Prossima lezione

- Continuiamo 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 è il primo/secondo/..."

S. Mizzaro - Programmazione strutturata 1

30