

Fondamenti di Informatica (117IN)

A.A. 2022 / 2023

Lezione 6 - Istruzioni e programma

Sylvio Barbon Junior
sylvio.barbonjunior@units.it



Sommario:

- 1) Istruzioni
 - a) Istruzioni condizionali
 - b) Istruzioni ripetitive
 - c) Istruzioni di salto

1) Ricordando

```
IMC.java
1 import java.io.Console;
2 import java.util.Scanner;
3
4 public class IMC{
5     public static void main (String args[]){
6         Console con = System.console();
7         Scanner sc = new Scanner(con.reader());
8
9         con.printf("*****\n");
10        con.printf("**Calcolo - Indice Massa Corpore (IMC) **");
11        con.printf("*****\n");
12
13        con.printf("Altezza: \n >>>");
14        double altezza = sc.nextDouble();
15
16        con.printf("Peso:\n >>>");
17        int peso = sc.nextInt();
18
19        con.printf("IMC = %.2f", (peso/(altezza*altezza)));
20    }
21 }
```

leggendo come double

leggendo come decimale

formatto con due cifre decimali

1) Ricordando



```
sylvioarbon@ws-piccola: ~/Scaricati/Fi/CodiciSorgenti
sylvioarbon@ws-piccola:~/Scaricati/Fi/CodiciSorgenti$ javac IMC2.java
sylvioarbon@ws-piccola:~/Scaricati/Fi/CodiciSorgenti$ java IMC2
*** Calcolo - Indice Massa Corpore (IMC) (V2)***
Altezza:
>>>1,93
Peso:
>>>98
Tuo IMC è 26,31, sei obeso.
sylvioarbon@ws-piccola:~/Scaricati/Fi/CodiciSorgenti$
```

1) Ricordando

conversione double per float

```
IMC2.java
1 import java.io.Console;
2 import java.util.Scanner;
3
4 public class IMC2{
5     public static void main (String args[]){
6         Console con = System.console();
7         Scanner sc = new Scanner(con.reader());
8         con.printf("*** Calcolo - Indice Massa Corpore (IMC) (V2)
9         con.printf("Altezza: \n >>>");
10        double altezza = sc.nextDouble();
11        con.printf("Peso:\n >>>");
12        int peso = sc.nextInt();
13
14        float imc = (peso/(float)(altezza*altezza));
15        con.printf("Tuo IMC è %.2f, sei ",imc);
16
17        if(imc<16){
18            con.printf("sottopeso grave");
19        }else if(imc<18.8){
20            con.printf("sottopeso");
21        }else if(imc<25){
22            con.printf("normale");
23        }else if(imc<30){
24            con.printf("sovrappeso");
25        }else{
26            con.printf("obeso");
27        }
28        con.printf("\n");
29    }
30 }
```

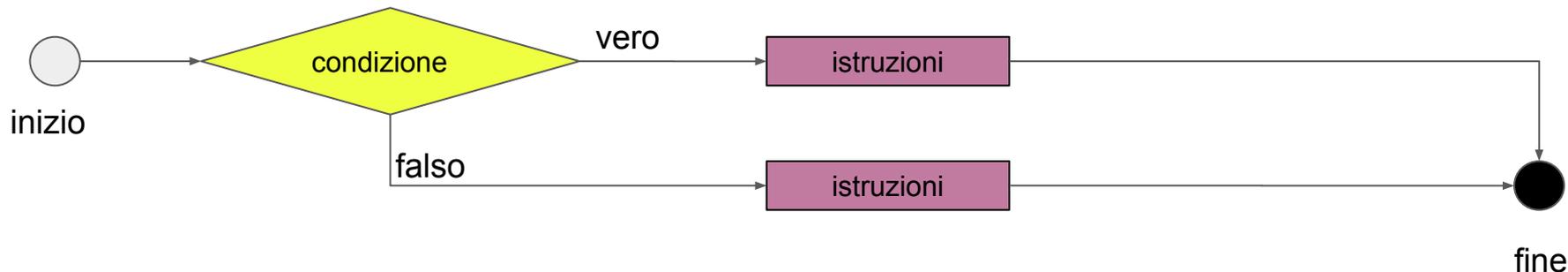
```
float imc = (peso/(float)(altezza*altezza));
con.printf("Tuo IMC è %.2f, sei ",imc);

if(imc<16){
    con.printf("sottopeso grave");
}else if(imc<18.8){
    con.printf("sottopeso");
}else if(imc<25){
    con.printf("normale");
}else if(imc<30){
    con.printf("sovrappeso");
}else{
    con.printf("obeso");
}
con.printf(".\n");
```

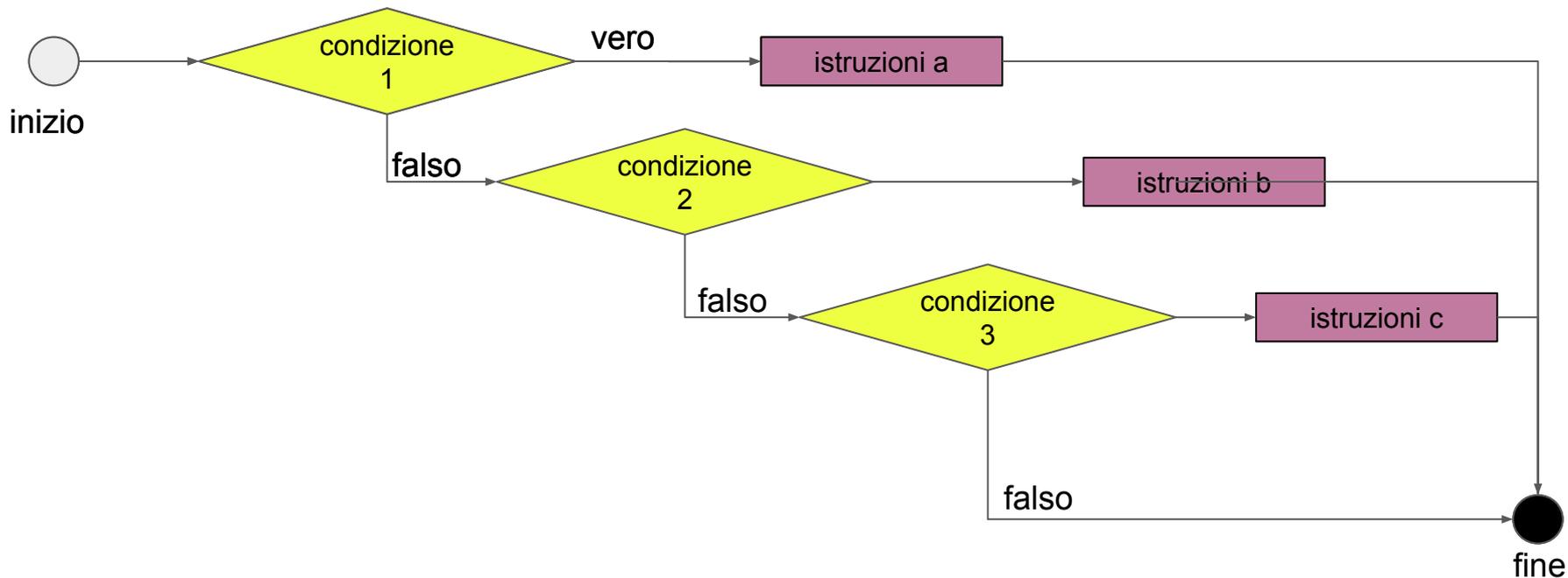
Le istruzioni
condizionali!

1) Istruzioni condizionali

- Le istruzioni condizionali comprendono l'istruzione **if** e l'istruzione **switch**.
- La **condizione** è un'espressione che deve produrre un risultato di tipo **booleano** (notare che non è possibile nessuna conversione di tipo tra intero e booleano, né implicita né indicata esplicitamente dal programmatore).
- La parte **else** può anche mancare.
- Se la condizione è vera, viene eseguita l'istruzione che segue **if**, altrimenti quella che segue **else** (se presente).



1) Istruzioni condizionali



1) Istruzioni condizionali

if - else

```
1 public class Radici{
2     public static void main(String[] args){
3         double a, b, c, delta;
4         a = Float.parseFloat(args[0]);
5         b = Float.parseFloat(args[1]);
6         c = Float.parseFloat(args[2]);
7         if (a==0 && b==0)
8             System.out.println("Equazione degenera");
9         else if (a==0){
10            System.out.println("Equazione di primo grado");
11            System.out.println(-c/b);
12        }
13        else{
14            delta = b*b - 4*a*c;
15            if (delta<0)
16                System.out.println("Determinante negativo");
17            else{
18                delta = Math.sqrt(delta);
19                System.out.println("Equazione di secondo grado");
20                System.out.println((-b+delta)/(2*a));
21                System.out.println((-b-delta)/(2*a));
22            }
23        }
24    }
25 }
```

prima condizione : {

seconda condizione : {

terza condizione : {

prima condizione della terza : }

seconda condizione della terza : }

1) Istruzioni condizionali

- Usando **switch**, la condizione deve essere un'espressione di tipo byte, char, short o int, ed ogni espressione costante (constant-expression), valutata a tempo di compilazione;
- La sequenza di istruzioni contenute nell'alternativa selezionata viene quindi eseguita:
 - l'ultima istruzione della sequenza deve produrre la terminazione dell'istruzione switch.
 - Tale terminazione può essere ottenuta con un'istruzione break.

```
1 public class NumeroPari{
2     public static void main (String args[]){
3         int numero = Integer.parseInt(args[0]);
4         switch(numero%2){
5             case 0:
6                 System.out.println("È pare"); } prima condizione
7                 break;
8             default:
9                 System.out.println("È dispare"); } seconda condizione
10                break;
11        }
12    }
13 }
```

1) Istruzioni condizionali

switch

```
DataSucc.java
1 public class DataSucc{
2     public static void main(String[] args){
3         int giorno, mese, anno, giornimese;
4         giorno = Integer.parseInt(args[0]);
5         mese = Integer.parseInt(args[1]);
6         anno = Integer.parseInt(args[2]);
7         switch (mese){
8             case 1: case 3: case 5: case 7: case 8: case 10: case 12:
9                 giornimese = 31;
10                break;
11             case 4: case 6: case 9: case 11:
12                 giornimese = 30;
13                 break;
14             case 2:
15                 if (anno % 4 == 0 && anno % 100 != 0 || anno % 400 == 0)
16                     giornimese = 29;
17                 else
18                     giornimese = 28;
19                 break;
20             default:
21                 System.out.println("Mese inesistente");
22                 giornimese = 0;
23                 System.exit(1);
24         }
25         if (giorno > giornimese)
26             System.out.println("Giorno inesistente");
27         else{
28             if (giorno < giornimese)
29                 giorno++;
30             else {
31                 giorno=1;
32                 if (mese<12)
33                     mese++;
34                 else {
35                     mese=1;
36                     anno++;
37                 }
38             }
39             System.out.println(giorno);
40             System.out.println(mese);
41             System.out.println(anno);
42         }
43     }
44 }
```

```
int giorno, mese, anno, giornimese;
giorno = Integer.parseInt(args[0]);
mese = Integer.parseInt(args[1]);
anno = Integer.parseInt(args[2]);
switch (mese){
    case 1: case 3: case 5: case 7: case 8: case 10: case 12:
        giornimese = 31;
        break;
    case 4: case 6: case 9: case 11:
        giornimese = 30;
        break;
    case 2:
        if (anno % 4 == 0 && anno % 100 != 0 || anno % 400 == 0)
            giornimese = 29;
        else
            giornimese = 28;
        break;
    default:
        System.out.println("Mese inesistente");
        giornimese = 0;
        System.exit(1);
}
```

1) Istruzioni condizionali

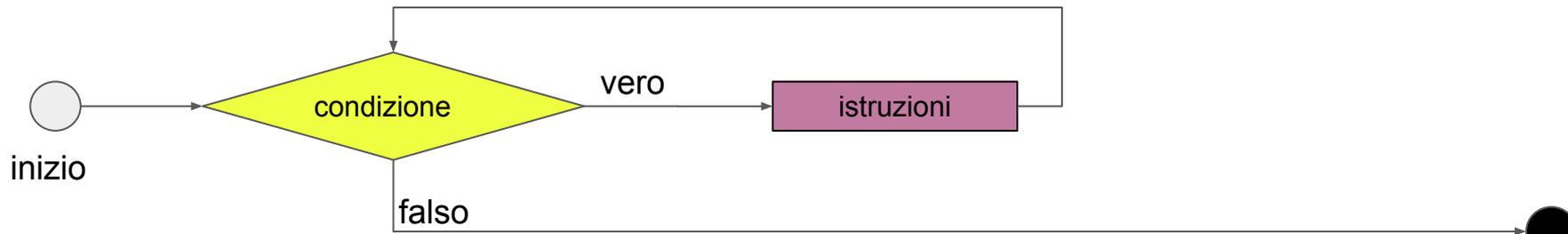
if - else

```
DataSucc.java
1 public class DataSucc{
2     public static void main(String[] args){
3         int giorno, mese, anno, giornimese;
4         giorno = Integer.parseInt(args[0]);
5         mese = Integer.parseInt(args[1]);
6         anno = Integer.parseInt(args[2]);
7         switch (mese){
8             case 1: case 3: case 5: case 7: case 8: case 10: case 12:
9                 giornimese = 31;
10                break;
11            case 4: case 6: case 9: case 11:
12                giornimese = 30;
13                break;
14            case 2:
15                if (anno % 4 == 0 && anno % 100 != 0 || anno % 400 == 0)
16                    giornimese = 29;
17                else
18                    giornimese = 28;
19                break;
20            default:
21                System.out.println("Mese inesistente");
22                giornimese = 0;
23                System.exit(1);
24        }
25        if (giorno > giornimese)
26            System.out.println("Giorno inesistente");
27        else{
28            if (giorno < giornimese)
29                giorno++;
30            else {
31                giorno=1;
32                if (mese < 12)
33                    mese++;
34                else {
35                    mese=1;
36                    anno++;
37                }
38            }
39            System.out.println(giorno);
40            System.out.println(mese);
41            System.out.println(anno);
42        }
43    }
44 }
```

```
if (giorno > giornimese)
    System.out.println("Giorno inesistente");
else{
    if (giorno < giornimese)
        giorno++;
    else {
        giorno=1;
        if (mese < 12)
            mese++;
        else {
            mese=1;
            anno++;
        }
    }
    System.out.println(giorno);
    System.out.println(mese);
    System.out.println(anno);
}
```

2) Istruzioni ripetitive

- Le istruzioni ripetitive comprendono l'istruzione **while**, l'istruzione **do-while** e l'istruzione **for**.
- **While** - La condizione è un'espressione che deve produrre un risultato di tipo booleano. Se la condizione è vera, viene eseguita l'istruzione che costituisce il corpo del costrutto e l'istruzione **while** viene ripetuta, altrimenti l'istruzione **while** termina.
- **Do-while**: viene valutata dopo che è stata eseguita l'istruzione che costituisce il corpo del costrutto, e, se vera, l'istruzione **do-while** viene ripetuta.
- **For**: l'inizializzazione è un'espressione di assegnamento che inizializza una variabile di controllo (eventualmente con definizione di tale variabile), e l'aggiornamento (o passo) è un'espressione che dà un nuovo valore alla variabile di controllo stessa (la incrementa o la decrementa).



2) Istruzioni ripetitive

```
MasComDiv.java x
1 public class MasComDiv{
2     public static void main(String[] args){
3         int alfa, beta, mcd;
4         alfa = Integer.parseInt(args[0]);
5         beta = Integer.parseInt(args[1]);
6         while (alfa != beta){
7             System.out.println(":alfa:"+alfa);
8             System.out.println(":beta:"+beta);
9             if (alfa > beta)
10                alfa -= beta;
11            else
12                beta -= alfa;
13        }
14        mcd = alfa;
15        System.out.println("***** MCD:"+mcd);
16    }
17 }
```

scopo dell while

```
sylviobarbon@ws-piccola:~/Scaricati/Fi/Codic
iSorgenti$ java MasComDiv 24 4
:alfa:24
:beta:4
:alfa:20
:beta:4
:alfa:16
:beta:4
:alfa:12
:beta:4
:alfa:8
:beta:4
***** MCD:4
sylviobarbon@ws-piccola:~/Scaricati/Fi/Codic
iSorgenti$
```

while

2) Istruzioni ripetitive

```
InputPari.java x
1  import java.io.Console;
2  import java.util.Scanner;
3  class InputPari{
4      public static void main(String[] args){
5          int c;
6
7          Console con = System.console();
8          Scanner sc = new Scanner(con.reader());
9
10         do{
11             con.printf("Immettere un numero pari:");
12             c = Integer.parseInt(sc.next());
13         }while(c % 2 != 0);
14         con.printf("Grazie! \n");
15     }
16 }
```

```
sylviobarbon@ws-piccol...
sylviobarbon@ws-piccola:~/Scaricati/FI/CodiciSorgen
ti$ java InputPari
Immettere un numero pari:565
Immettere un numero pari:879
Immettere un numero pari:7
Immettere un numero pari:13
Immettere un numero pari:2
Grazie!
sylviobarbon@ws-piccola:~/Scaricati/FI/CodiciSorgen
ti$
```

do-while

2) Istruzioni ripetitive

```
Fatt.java
1  public class Fatt{
2      public static void main(String[] args){
3          int num, fattoriale = 1;
4          num = Integer.parseInt(args[0]);
5          for (int i = 2; i <= num; i++){
6              System.out.println(fattoriale);
7              fattoriale *= i;
8          }
9          System.out.println(fattoriale);
10     }
11 }
```

```
sylvioarbon@ws-piccola: ...
sylvioarbon@ws-piccola:~/Scaricati/FI/CodiciSorgenti
$ java Fatt 8
1
2
6
24
120
720
5040
40320
sylvioarbon@ws-piccola:~/Scaricati/FI/CodiciSorgenti
$
```

for (inizio, fine, incremento)

3) Istruzioni di salto

- Un'istruzione di salto cambia il flusso di ripetizioni di esecuzioni dei codici.

Istruzione break:

Produce la terminazione del costrutto switch, while, do-while o for in cui compare

Istruzione continue:

Produce la terminazione della iterazione corrente del ciclo while, do-while o for in cui compare, con esecuzione della iterazione successiva

Istruzione return:

Produce la terminazione della funzione in cui compare.

3) Istruzioni di salto

```
EsempioBreak.java x
1 public class EsempioBreak{
2     public static void main(String args[]){
3         for(int i=1; i<=10; i++){
4             System.out.println("i:"+i);
5             if(i==5){
6                 break;
7             }
8         }
9         System.out.println("Fine!");
10    }
11 }
```

```
sylvioarbon@ws-piccola: ...
sylvioarbon@ws-piccola:~/Scaricati/FI/CodiciSorgenti
$ javac EsempioBreak.java
sylvioarbon@ws-piccola:~/Scaricati/FI/CodiciSorgenti
$ java EsempioBreak
i:1
i:2
i:3
i:4
i:5
Fine!
sylvioarbon@ws-piccola:~/Scaricati/FI/CodiciSorgenti
$
```

break

3) Istruzioni di salto

continue

```
EsempioCont.java x
1 public class EsempioCont{
2     public static void main(String args[]){
3         for(int i=1; i<=10; i++){
4             if(i==5){
5                 continue;
6             }
7             System.out.println("i:"+i);
8         }
9         System.out.println("Fine!");
10    }
11 }
```

```
sylvioarbon@ws-piccola: ...
$ javac EsempioCont.java
sylvioarbon@ws-piccola:~/Scaricati/Fi/CodiciSorgenti
$ java EsempioCont
i:1
i:2
i:3
i:4
i:6
i:7
i:8
i:9
i:10
Fine!
sylvioarbon@ws-piccola:~/Scaricati/Fi/CodiciSorgenti
$
```

3) Istruzioni di salto

return

```
EsempioReturn.java x
1 public class EsempioReturn{
2     public static void main(String args[]){
3         for(int i=1; i<=10; i++){
4             System.out.println("i:"+i);
5             if(i==5){
6                 return;
7             }
8         }
9         System.out.println("Fine!");
10    }
11 }
```

```
sylvioarbon@ws-piccola: ...
Fine!
sylvioarbon@ws-piccola:~/Scaricati/FI/CodiciSorgenti
$ javac EsempioReturn.java
sylvioarbon@ws-piccola:~/Scaricati/FI/CodiciSorgenti
$ java EsempioReturn
i:1
i:2
i:3
i:4
i:5
sylvioarbon@ws-piccola:~/Scaricati/FI/CodiciSorgenti
$
```



Grazie!!