

# Programmazione I

A.A. 2002-03

---

## Costrutti di base

( Lezione XIII )

### Costrutto di iterazione “while”

---

**Prof. Giovanni Gallo**

**Dr. Gianluca Cincotti**

Dipartimento di Matematica e Informatica

Università di Catania

e-mail : { [gallo](mailto:gallo@dmf.unict.it), [cincotti](mailto:cincotti@dmf.unict.it) } @[dmf.unict.it](mailto:dmf.unict.it)

## *Notazione Lineare Strutturata*

---

➤ Tre costrutti fondamentali :

- Sequenza,
- Selezione,
- Iterazione.

➤ Servono a controllare il flusso del programma !

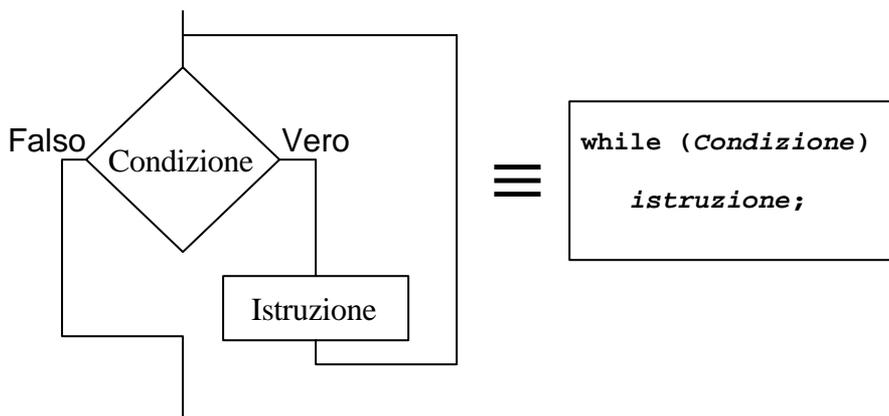
## *Istruzioni iterative*

---

- I costrutti di *iterazione* consentono di eseguire molte volte la stessa istruzione e sono controllati da espressioni booleane.
- In Java esistono *tre* tipi di istruzioni iterative:
  - il ciclo *while*,
  - il ciclo *do...while*,
  - il ciclo *for*.
- Sono equivalenti ma...
  - appropriati in situazioni diverse !

## *Ciclo “while”*

---



# Esempi

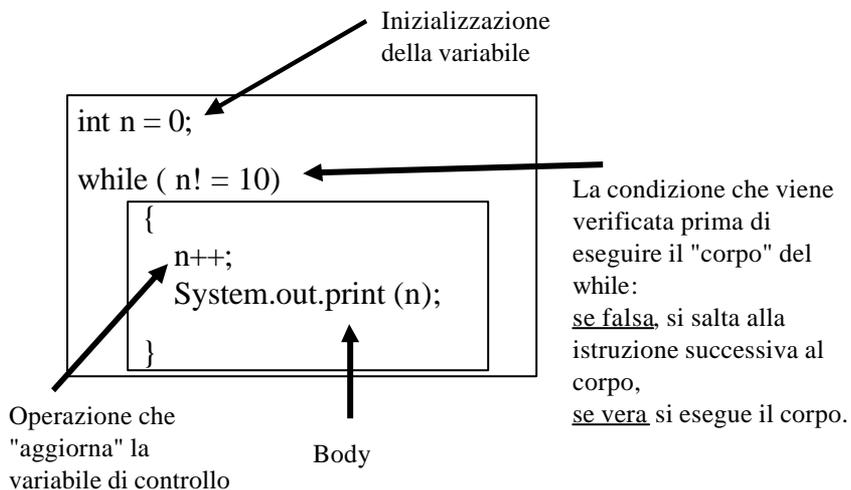
```
int n = 0;
while ( n! = 10)
    System.out.print (n++);
```

Il "corpo" del *while* è costituito da una singola istruzione.

```
int n = 0;
while ( n! = 10)
{
    n++;
    System.out.print (n);
}
```

Il "corpo" del *while* è costituito da un "blocco" di istruzioni.

# Al microscopio !



## *Caratteristica del “while”*

---

➤ Se la condizione di un ciclo *while* è inizialmente *falsa*, il ciclo non viene **mai** eseguito.

- Quindi un ciclo *while* può essere eseguito 0 o più volte.

## *Esempio*

---

```
int n = 10;
while ( n! = 10)
{
    n++;
    System.out.print (n);
}
```

La condizione è sempre falsa!

## Ciclo “while” infinito

---

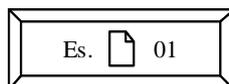
- Il corpo di un ciclo *while* deve alla fine rendere *falsa* la condizione perchè il ciclo si fermi.
- Altrimenti è un *ciclo infinito* (loop), che viene eseguito finché l’utente non interrompe il programma
  - Costituisce un tipico errore logico molto frequente.
    - Assicuratevi che i vostri cicli abbiano sempre termine !

## Esempio

---

```
int n = 0;
while ( n! = 10)
{
    System.out.print (n);
}
```

La condizione sarà sempre vera!



## *Cicli annidati*

---

- Anche i cicli possono essere annidati.
  - Come le istruzioni *if...else*
- Cioè, il corpo del ciclo contiene un altro ciclo.
- Ogni nuovo ingresso nel ciclo esterno causa *un'intera* esecuzione del ciclo interno.

## *Un esempio più complesso ma usuale: while annidati*

---

```
1  int i=0;
2  while(i<10)
3      {
4          j=10;
5          while(j>0)
6              {
7                  System.out.println("i="+i+"e j="+j);
8                  j--;
9              }
10         i++;
11     }
```

Es.  03

---

*Fine*