

# COMANDO CONDIZIONALE (IF)

lunedì 18 settembre 2017 11:18

SINTASSI:

if ( ESPRESSIONE LOGICA )

Comando

GUARDIA /  
CONDIZIONE

DETTI  
"RAMI"  
DELL'IF

else  
Comando

SE LA  
CONDIZIONE  
E' VERA  
ESECUCE  
QUESTO

SE LA CONDIZIONE  
E' FALSA ESECUCE  
QUESTO

E' OPZIONALE

ESEMPIO: CALCOLO DEL MAX TRA DUE NUMERI

```
int x = 10;
```

```
int y = 12;
```

y = 8;

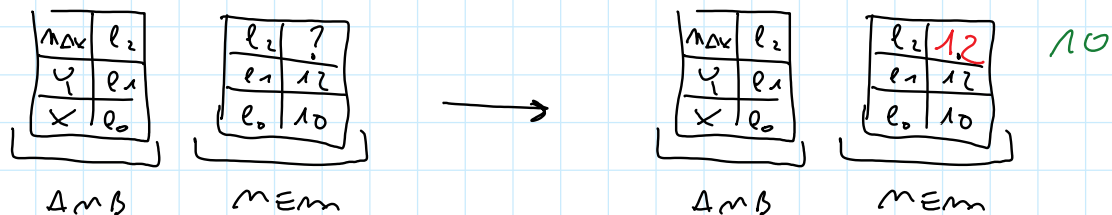
```
int max;
```

```
if ( x > y )
```

```
    max = x;
```

```
else
```

```
    max = y;
```



CAMBIANDO I VALORI INIZIALI  
DELLE VARIABILI CAMBIA  
IL COMPORTAMENTO

# ESPRESSIONI LOGICHE

INCLUDONO:

- ESPRESSIONI di CONFRONTO

$>$ ,  $<$ ,  $>=$ ,  $<=$ ,  $==$ ,  $!=$

DIVERSO

$\uparrow$   $\uparrow$   $\uparrow$   
MAGGIORE o UGUALE minore o UGUALE UGUAGLIANZA

ESEMPI :

$x > 0$        $x <= 0$        $x == y$

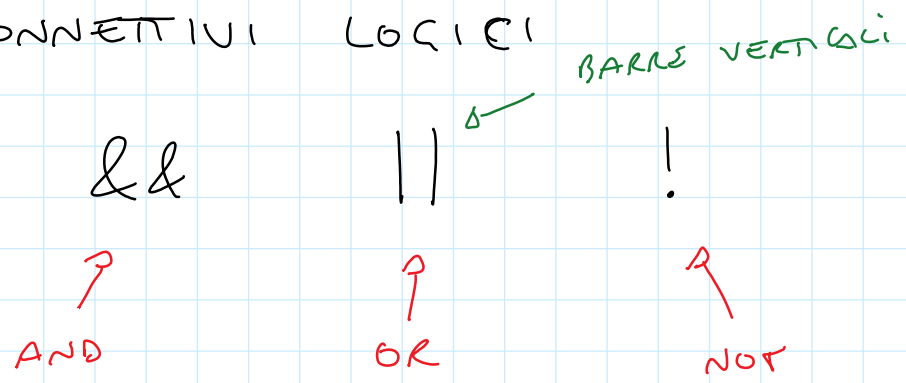
$x + y > 0$        $x + 3 != y + 2$

ATTENZIONE : UGUAGLIANZA SI FA

CON  $==$ , NON

CON  $=$  (CHE È ASSEGNAMENTO)

# - CONNETTIVI LOGICI



ESEMPIO: VARIABILE X CHE VOGLIAMO MANTENERE ENTRO UN CERTO INTERVALLO  $[-10, 10]$

SE IL VALORE DI X È FUORI DALL'INTERVALLO, RIAZZERO X

```
int x = 12;  
if ( (x < -10) || (x > 10) )  
    x = 0;
```

ESEMPIO: ASSEGNO 1 A Z SE  
X e Y SONO CONCORDI,  
ASSEGNO -1 A Z SE  
INVECE SONO DISCORDI

```
int x = 10;
```

```
int y = -2;
```

```
int z;
```

```
if (((x > 0) && (y > 0)) ||  
    ((x < 0) && (y < 0)))
```

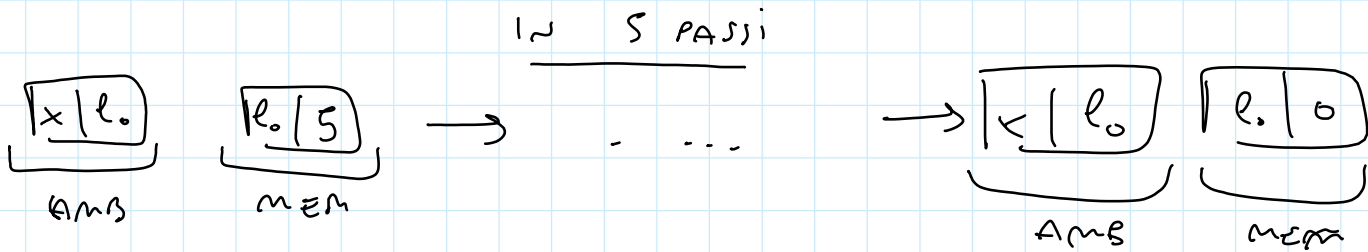
```
    z = 1;
```

```
else
```

```
    z = -1;
```

# ESEMPIO di COMANDI RIPETUTI

→ CONTO ALLA ROVERSCIA



```
int x = 5 6  
x = x - 1;  
x = x - 1;  
x = x - 1;  
x = x - 1;  
x = x - 1;
```

} NO ...

# COMANDO ITERATIVO (CICLO)

SINTASSI: while ( ESPRESSIONE LOGICA )  
          comando

← GUARDIA

← QUESTO comando  
SARA' RIPETUTO  
FINIANTO CHE  
LA CONDIZIONE  
ESPRESSA DALLA  
GUARDIA E'  
VERA

→ ESEMPIO : CONTO ALLA ROVESCIA

```
int x = 5;
while (x > 0)
  x = x - 1;
```

← 6

IN QUESTO  
CASO IL PROGRAMMA  
FUNZIONA PER  
TUTTI I VALORI  
DI x maggiori  
di ZERO

- ESEMPIO: CA DEL FATTORIALE  
di UN NUMERO.

IN  
MATEMATICA  
SI RAPPRESENTA  
CON !

$$5! = 1 * 2 * 3 * 4 * 5$$

$$m! = 1 * \dots * m$$

```

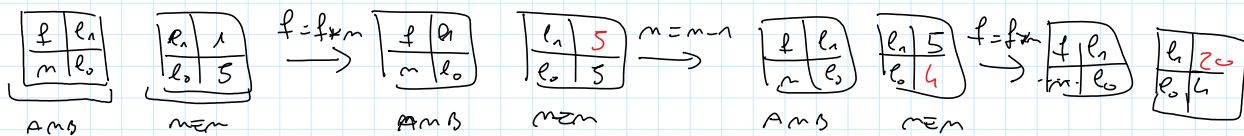
int m = 5;
int f = 1;
while (m > 1) {
    f = f * m;
    m = m - 1;
}
    
```

CORPO DEL WHILE

IDEA:  
ESEGUIAMO  
LE MOLTIPLICAZIONI  
IN ORDINE  
INVERSO ...  
DAL PIU' GRANDE  
AL PIU' PICCOLO  
5 \* 4 \* 3 \* 2 \* 1

QUESTE DUE PARENTESI  
MI RAGGRUPPANO IL  
CONTENUTO CREANDO UN  
UNICO COMANDO

### BLOCCO



RIPETUTI  
FINANTO CHE  $m > 1$



# ESEMPIO: CALCOLO MCD CON METODO DI EUCLIDE

IDEA: DATI  $x$  E  $y$

SOTTRAGGO IL PIU' PICCOLO  
DAL PIU' GRANDE

CONTINUO FINO A CHE I DUE  
NUMERI DIVENTANO UGUALI

QUELLO SARA' IL MCD

$x$	$y$
12	30
12	18
12	6
6	6

← SONO UGUALI  
MCD = 6

```
int x = 30;  
int y = 12;  
int mcd;  
while (x != y)  
    if (x > y)  
        x = x - y;  
    else  
        y = y - x;  
mcd = x;
```

TUTTO QUESTO È  
UN UNICO  
COMANDO CONDIZIONALE

← QUESTO ASSEGNAMENTO  
VIENE ESEGUITO  
AL TERMINE  
DEL CICLO

DOMANDA :

QUESTO PROGRAMMA TERMINA?

RISPOSTA :

SI SE  $x > 0$  e  $y > 0$

MCD È UN ESEMPIO DI  
ITERAZIONE INDETERMINATA  
(NON SO QUANTE VOLTE IL CICLO  
VIENE RIPETUTO)

FATTORIALE È UN ESEMPIO DI  
ITERAZIONE DETERMINATA  
(SO QUANTE VOLTE VERRÀ RIPETUTO  
IL CICLO ...  $n-1$  VOLTE)

# COMANDO PER ITERAZIONE DETERMINATA (FOR)

SINTASSI: `for ( comando ; espressione ; comando )`

INIZIALIZZAZIONE ↓  
GUARDIA ↓  
AGGIORNAMENTO ↓

comando ← CORPO

ESEMPIO : FATTORIALE

```

int m = 5; int f = 1;
for ( int i = 1; i <= m; i = i + 1 )
  f = f * i;
  
```

DICHIAZIONE  
CON  
INIZIALIZZAZIONE  
DI UNA  
VARIABILE  
CHE USO PER  
CONTARE IL  
NUMERO DI  
VOLTE CHE  
RIPETO IL  
CICLO

ALTRE FORME  
DI ASSEGNAMENTO

$i = i + 1 \Leftrightarrow i++$

$i = i - 1 \Leftrightarrow i--$

$x = x + 5 \Leftrightarrow x += 5$

LI VEDRETE IN LAB.

## IL CICLO FOR :

- 1) ESEGUE L'INIZIALIZZAZIONE  
(int i = 1)
- 2) VALUTATA LA CONDIZIONE
  - ↳ SE È VERA
    - 2a) ESEGUE IL CORPO
    - 2b) ESEGUE IL COMANDO DI AGGIORNAMENTO
    - 2c) RITORNA DA 2)
  - ↳ SE È FALSA  
TERMINA

# ULTIMA OSSERVAZIONE SUI BLOCCHI

mercoledì 20 settembre 2017 17:44

UN BLOCCO { ... } CONSENTE DI  
CONSIDERARE UNA SEQUENZA DI  
COMANDI COME UN UNICO COMANDO

```
if (...) {  
    ⋮  
    ← SEQ. di COMANDI  
}
```

```
if (x > 0) {  
    x = x - 1;  
    y = y - 1;  
}
```

```
while (...) {  
    ⋮  
}
```

```
for (...) {  
    ⋮  
}
```

COSSA SUCCEDERE SE IN BLOCCO  
INSERISCO UNA DICHIARAZIONE  
DI VARIABILE?

```
int x = 8;  
if (x > 0) {  
    int y = 2;  
    x = x - y;  
}
```

```
x = x + y;
```

ERRORE!  
fuori

COSSA SUCCEDERE  
A QUESTA  
VARIABILE?

↓  
PUO' ESSERE  
UTILIZZATA  
SOLO ALL'INTERNO  
DEL BLOCCO  
STESSO