

# Il vero e il falso

Informatica@DSS 2025/2026

Massimo Lauria <massimo.lauria@uniroma1.it>  
<https://massimolauria.net/informatica2025/>

# Prendere decisioni

# Scegliere le istruzioni da eseguire

```
# Dati in input
```

```
prezzo = 280
```

```
sconto = -18
```

```
if sconto < 0:
```

```
    print("Uno sconto del",sconto,"percento non è valido")
```

```
print("Fine programma")
```

1  
2  
3  
4  
5  
6  
7  
8  
9

```
Uno sconto del -18 percento non è valido
```

```
Fine programma
```

## Scegliere le istruzioni da eseguire (2)

<i># Dati in input</i>	1
prezzo = 280	2
sconto = 18	3
	4
if sconto < 0:	5
print("Uno sconto del",sconto,"percento non è valido")	6
	7
print("Fine programma")	8

Fine programma

# Clausola if

La clausola `if` testa se un'espressione è vera, ad esempio  $2*x > 3.4$ , e se lo è esegue il **blocco di codice** sottostante.

<code>if condizione:</code>	1
<code>istruzione1</code>	2
<code>istruzione2</code>	3
<code>istruzione3</code>	4
	5
<code>istruzione4</code>	6
<code>istruzione5</code>	7

- ▶ Se condizione è vera: vengono eseguire tutte e 5 le istruzioni.
- ▶ Se condizione è falsa: vengono eseguire le istruzioni 4 e 5.

# Sintassi della clausola if

if condizione:	1
istruzione1	2
istruzione2	3
istruzione3	4
	5
istruzione4	6
istruzione5	7

Notare la sintassi

- ▶ i due punti dopo espressione
- ▶ la spaziatura allineata sotto la clausola if

Fanno parte del **blocco di codice** if

- ▶ istruzione1 deve essere **indentata**
- ▶ tutte le istruzioni indentate al suo stesso livello

# Esempio di clausola if

temperatura = 124	1
	2
if temperatura <= 0:	3
print("Stato solido")	4
print("Abbiamo del ghiaccio.")	5
	6
if temperatura >= 100:	7
print("Stato aeriforme.")	8
print("Siamo sopra al punto di ebollizione.")	9

Stato aeriforme.  
Siamo sopra al punto di ebollizione.

# Espressioni booleane: True e False

```
print( type(True) )           1
print( type(False) )         2
bocciato = False               3 # variabile inizializzata a False
print( type(bocciato) )       4
print( False )               5
print( True )                6
print( false )               7 # False con l'iniziale maiuscola
```

```
<class 'bool'>
<class 'bool'>
<class 'bool'>
False
True
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "/tmp/babel-7n28gk/python-nvS0VM", line 7, in <module>
    print( false )           # False con l'iniziale maiuscola
NameError: name 'false' is not defined
```



# Operatori di confronto `expr1 op expr2`

```
print( type( 3 < 0 ) )      1
print( 3 < 0 )              2
print( 3.4 > -2 )          3
print( 5 >= 5 )             4  # maggiore o uguale?
print( 5 <= -12 )           5  # minore o uguale?
print( 'Gatto' != 'gatto' ) 6  # sono diversi?
print( 'Gatto' == 'gatto' ) 7  # sono uguali?
print( 3 == 4 )             8  # sono uguali?
```

```
<class 'bool'>
False
True
True
False
True
False
False
```

# Operatori logici

Negazione: `not` X vero solo quando X è falso

Congiunzione: X `and` Y vero quando entrambi veri

Disgiunzione: X `or` Y vero quando **almeno** uno è vero

```
print (not True)           1
X = False                  2
print (not X)              3
                             4
Y = False                  5
print (X or not Y)         6
                             7
print (True or True)       8
                             9
print(not X and True)     10
```

```
False
True
True
True
True
```

# Esempio

```
pioggia = False      1
nuvoloso = True      2
if pioggia or nuvoloso: 3
    print("1. Prenderò l'ombrello") 4
    print("1. Prenderò le scarpe chiuse") 5
6
nuvoloso = False      7
if pioggia or nuvoloso: 8
    print("2. Prenderò l'ombrello") 9
    print("2. Prenderò le scarpe chiuse") 10
```

```
1. Prenderò l'ombrello
1. Prenderò le scarpe chiuse
```

# Esempio

```
sconto = -15      1
prezzo = 240      2

if sconto < 0 or sconto > 100:      3
    print("Uno sconto del",sconto,"percento non è valido")      4
                                                                    5
                                                                    6
prezzofinale = prezzo*(100-sconto)/100      7
print("Prezzo scontato: ",prezzofinale)      8
                                                                    9
```

```
Uno sconto del -15 percento non è valido
Prezzo scontato:  276.0
```

Oltre la clausola if

# Due alternative (If-else)

```
pioggia = False      1
nuvoloso = False     2
if pioggia or nuvoloso: 3
    print("Prenderò l'ombrello") 4
else:                  5
    print("Prenderò i sandali") 6
```

Prenderò i sandali

# Sintassi del costrutto if else

```
if condizione:                                1
    blocco1                                   2
    blocco1                                   3
    blocco1                                   4
else:                                          5
    blocco2                                   6
    blocco2                                   7
```

oppure (anche se fa un po' schifo)

```
if condizione:                                1
    blocco1                                   2
    blocco1                                   3
    blocco1                                   4
    blocco1                                   5
else:                                          6
    blocco2                                   7
    blocco2                                   8
```

# Esempio

```
sconto = -15 1
prezzo = 240 2
3
if sconto < 0 or sconto > 100: 4
    print("Uno sconto del",sconto,"percento non è valido") 5
else: 6
    prezzofinale = prezzo*(100-sconto)/100 7
    print("Prezzo scontato: ",prezzofinale) 8
9
```

Uno sconto del -15 percento non è valido



# Esercizio

Scriviamo un programma che dati due intervalli  $[a, b]$  e  $[c, d]$  decide se uno è contenuto nell'altro.

```
# Dati in input
a = 10
b = 20
c = 7
d = 23

if (a <= c and d <= b) or (c <= a and b <= d):
    print("Un intervallo è contenuto nell'altro.")
else:
    print("Gli intervalli non sono confrontabili.")
```

Un intervallo è contenuto nell'altro.

# Annidamento di clausole

```
voto=20  
  
print("Il voto e'", voto)  
if voto < 18:  
    print("mi dispiace")  
else:  
    if voto == 18:  
        print("appena sufficiente")  
    else:  
        if voto < 24:  
            print("OK, ma potevi fare meglio")  
        else:  
            if voto == 30:  
                print("congratulazioni!")  
            else:  
                print("bene!")
```

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16

```
Il voto e' 20  
OK, ma potevi fare meglio
```

# Altro annidamento

```
voto=26 1
2
print("Il voto e'", voto) 3
if voto < 24: 4
    if voto<=18: 5
        if voto==18: 6
            print("appena sufficiente") 7
        else: 8
            print("mi dispiace") 9
    else: 10
        print("OK, ma potevi fare meglio") 11
else: 12
    if voto == 30: 13
        print("congratulazioni!") 14
    else: 15
        print("bene!") 16
```

```
Il voto e' 26
bene!
```

# Blocchi annidati

Ad ogni clausola `if` o `else`

- ▶ si deve aumentare di un livello l'intentazione (lo spazio prima dell'inizio della riga)
- ▶ le righe nel medesimo blocco devono avere la stessa indentazione.

# Clausola elif

elif è un'abbreviazione di else if

```
voto= 20 1
2
print("Il voto e' "+ str(voto) + ".") 3
if voto < 18: 4
    print("Mi dispiace...") 5
elif voto == 18: 6
    print("Appena sufficiente.") 7
elif voto < 24: 8      # tra 19 e 23
    print("OK, ma potevi fare meglio.") 9
elif voto == 30: 10
    print("Congratulazioni!") 11
else: 12      # tra 24 e 29
    print("Bene!") 13
```

Le condizioni vengono testate **a cascata**. E.g. il test nella linea 7 viene effettuato solo se quelli alle linee 3 e 5 sono falliti.

# Esempio: contiamo le soluzioni

$$Ax^2 + Bx + C = 0$$

```
1  # Dati
2  A = 1.0
3  B = 12
4  C = 3
5
6  Delta = B**2 - 4*A*C
7
8  if A==0:
9      print("Non è un'equazione propria di 2o grado")
10
11 elif Delta < 0:
12     print("Nessuna soluzione reale")
13
14 elif Delta == 0:
15     print("Una soluzione reale")
16
17 else:
18     print("Due soluzioni reali")
```

## Esercizio:

Scrivere un programma che stampi i tre numeri nelle tre variabili A ,B ,C dal più piccolo al più grande.

# Altro su operatori e espressioni



# Precedenze di operatori

## 1. Aritmetici

- \*\* (unico valutato da destra a sinistra)
- segni + e - (per esempio -2 e +2.4)
- /, //, %, \*
- +, -

## 2. Confronti (stessa precedenza)

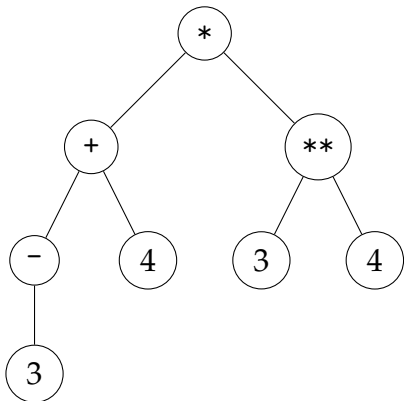
- <, >, <=, >=, ==, !=

## 3. Logici

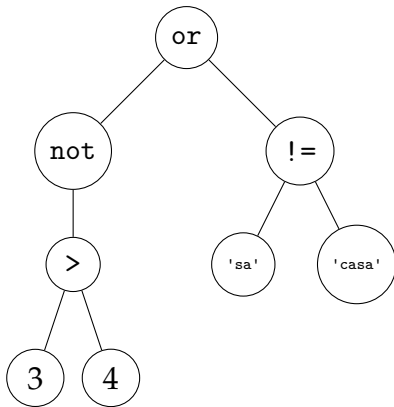
- not prima di and prima di or

Gli altri operatori sono nella **documentazione**

`(- 3 + 4) * 3**4`



`not 3 > 4 or 'sa' != 'casa'`



# Uguaglianza e assegnamenti

- ▶ L'operatore == determina se due operandi sono uguali
- ▶ Il simbolo = indica un assegnamento di variable

<code>variabile = "valore assegnato"</code>	1
<code>variabile == "altra stringa"</code>	2
<code>print(variabile)</code>	3
<code>print(variabile == "altra stringa")</code>	4

<code>valore assegnato</code>
<code>False</code>

# Catene di confronti

In python è possibile scrivere

```
a1 op1 a2 op2 a3 op3 ... aN
```

dove op1, op2,... sono operatori di confronto. Ad esempio

```
N=3.4 1
if 0 < N < 10: 2
    print("Nell'intervallo di sicurezza") 3
```

è “equivalente” a

```
N=3.4 1
if 0 < N and N < 10: 2
    print("Nell'intervallo di sicurezza") 3
```

# Altri esempi di catene di confronti

<code>x = 5</code>		1
<code>print(1 &lt; x &lt; 10)</code>	<code># 1 &lt; x and x &lt; 10</code>	2
<code>print(10 &lt; x &lt; 20 )</code>	<code># 10 &lt; x and x &lt; 20</code>	3
<code>print(x &lt; 10 &lt; x*10 &lt; 100)</code>	<code># x&lt;10 and 10&lt; x*10 and x*x&lt;100</code>	4
<code>print(10 &gt; x &lt;= 9)</code>	<code># 10 &gt; x and x &lt;= 9</code>	5
<code>print(5 == x &gt; 4)</code>	<code># 5 == x and x&gt;4</code>	6

True  
False  
True  
True  
True

# Esercizio: calcolare la seguente espressione

```
not -5//2**4 < -1 and 3 ** 2 ** (5 + - 3) >= 2*4
```

1