

Il ciclo While

Scrivi ed esegui i seguenti programmi:

Esempio 1

```
a = 0
while a < 10:
    a = a + 1
    print a
```

Esempio 2

```
a = 0
while a < 10:
    a = a + 1
print a
```

Cosa noti? Ricorda che l'indentazione in Python è molto importante!

Esempio 3

```
a = 0
while a < 10:
    a = a + 1
    print a,
```

Cosa noti?

Esempio 4

```
# Questo programma calcola la sequenza di Fibonacci.
a = 0
b = 1
count = 0
max_count = 20
while count < max_count:
    count = count + 1
    # Occorre tenere traccia finché ci sono cambiamenti
    old_a = a
    old_b = b
    a = old_b
    b = old_a + old_b
    # Attenzione che la virgola alla fine di un'istruzione print
    # prosegue la stampa sulla stessa linea.
    print old_a,
print
```

Esempio 5

```
# Attende sino a quando non viene inserita la giusta password.
# Usate Control-C per fermare il programma senza password.
# Notate che se non viene inserita la giusta password, il ciclo
# while prosegue all'infinito.
password = "foobar"
# Notate il simbolo != (diverso).
while password != "unicorn":
    password = raw_input("Password:")
print "Benvenuto"
```

L'istruzione if

Scrivi ed esegui il seguente programma:

```
n = input("Inserisci un numero: ")
if n < 0:
    print "Il valore assoluto di",n,"e'",-n
else:
    print "Il valore assoluto di",n,"e'",n
```

Python decide se l'affermazione `n < 0` è vera o falsa. Un'istruzione `if` è sempre seguita da un *blocco* di altre istruzioni indentate che vengono eseguite nel caso l'affermazione sia vera. In caso contrario vengono eseguiti i comandi indentati sotto l'istruzione `else`, ovvero quando l'affermazione `if` risulta falsa.

Python presenta svariati operatori associabili al confronto di una variabile, ecco la lista completa:

Operatore	Funzione
<	minore di
<=	minore o uguale a
>	maggiore di
>=	maggiore o uguale a
==	uguale
!=	diverso da
<>	variante di diverso da

Un'altra proprietà dell'istruzione `if` è la funzione `elif`. È un'abbreviazione di `else if` e significa: esegui il blocco di istruzioni tabulato sotto `elif` se la condizione necessaria al primo `if` è falsa e la condizione `elif` è vera.

Ecco un esempio (scrivi, salva ed esegui il programma: attento alle indentazioni! Aggiungi qualche riga di commento)

```
a = 0
while a < 10:
    a = a + 1
    if a > 5:
        print a," > ",5
    elif a <= 7:
        print a," <= ",7
    else:
        print "Scrivi tu una frase appropriata"
```

Altro esempio

```
# Giocate ad indovinare il numero alto-basso
number = 78
indovinato = 0
while indovinato != number :
    indovinato = input ("Indovina un numero: ")
    if indovinato > number :
        print "Troppo grande"
    elif indovinato < number :
        print "Troppo piccolo"
print "Esatto! Il numero da indovinare era ", number
```

Esercizio 2

Scrivi un programma che permetta di stabilire se un numero intero è pari o dispari.

Esercizio 3

Scrivi un programma che chieda all'utente di indovinare una password, ma che dia al giocatore solamente 3 possibilità, fallite le quali terminerà, stampando "E' troppo complicato per voi!".