

Python 202

“giuliof”

GOLEM

20 giugno 2018



Cosa abbiamo fatto la volta scorsa?

- Stampa su schermo (comando `print`);
- Operazioni logiche e matematiche;
- Variabili e stringhe;
- Lettura di valori (`input` e `raw_input`);
- Il blocco condizionale (`if`, `elif`, `else`);

Link alla lezione precedente:

https://golem.linux.it/wiki/Ore_del_GOLEM#22_Maggio:_Python_-_introduzione_e_basi

Esercizio per casa: il quizzone

- Realizzare un programma che ponga, una dopo l'altra, 3 domande e ne verifichi la risposta
- Al termine viene stampato il punteggio, +1 per ogni risposta corretta, -0.5 per ogni risposta sbagliata

Raggruppare il codice: le funzioni

```
def chiedimi(domanda,risposta,nuovo_punteggio): ## Questa e' una funzione
    mia_risposta = raw_input(domanda)
    if risposta == mia_risposta:
        print "Corretto!"
        nuovo_punteggio = nuovo_punteggio + 1
    else:
        print "Sbagliato!"
        # Forma abbreviata
        nuovo_punteggio -= 0.5
    return nuovo_punteggio

## Il programma inizia da qui
punteggio = 0.0
punteggio = chiedimi("Di_che_colore_era_il_cavallo_bianco_di_Napoleone_", "
    ↪ Bianco", punteggio)
punteggio = chiedimi("Quanti_trentini_entrarono_a_Trento_", "Trenta", punteggio)
punteggio = chiedimi("Quanto_fa_uno_piu'_uno", "due", punteggio)
print "hai_fatto_%f_punti" % punteggio
```

Esercizio: rettangolo

Trovare se un punto è all'interno di un rettangolo con vertici:

$$A = (0, 10)$$

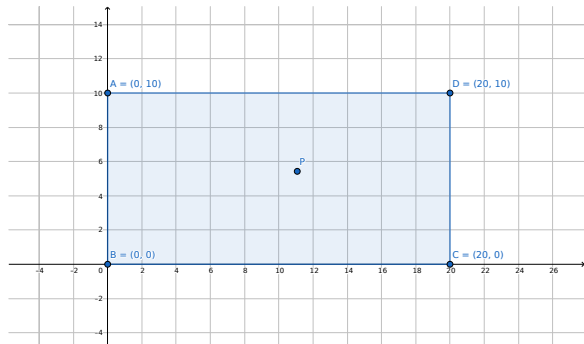
$$B = (20, 10)$$

$$C = (0, 0)$$

$$D = (20, 0)$$

Le coordinate del punto si inseriscono da tastiera.

Scrivere una funzione che, date le coordinate xy del punto, restituisca 1 oppure 0, vero oppure falso...



Esercizio: rettangolo

Trovare se un punto è all'interno di un rettangolo con vertici:

$$A = (0, 10)$$

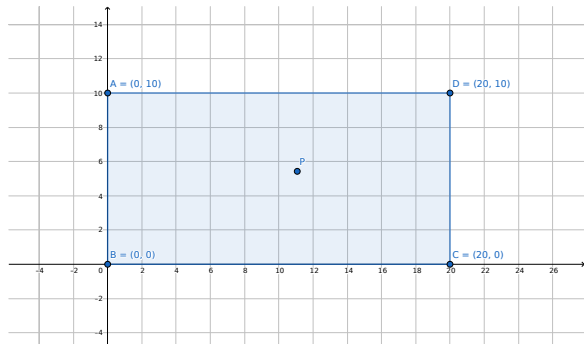
$$B = (20, 10)$$

$$C = (0, 0)$$

$$D = (20, 0)$$

Le coordinate del punto si inseriscono da tastiera.

Scrivere una funzione che, date le coordinate xy del punto, restituisca 1 oppure 0, vero oppure falso...



Soluzione “a marrano”: rettangolo

```
def inRettangolo(x, y):  
    if ((x < 20) and (x > 0)) and ((y < 10) and (y > 0)):  
        return True  
    else:  
        return False
```

Segue poi il programma che chiede x e y in input e poi chiama la funzione, stampando a video la risposta.

Soluzione “perbenino”: rettangolo

```
def inRettangolo(x, y):
    x_minima = 0
    x_massima = 20
    y_minima = 0
    y_massima = 10

    # ipotizzo subito che il punto sia dentro al rettangolo
    dentro = True

    # guardo se e' fuori dalle x
    if (x > x_massima) or (x < x_minima):
        dentro = False # o anche return False
    # guardo se e' fuori dalle y
    if (y > y_massima) or (y < y_minima):
        dentro = False # o anche return False

    return dentro
```


Ripetere le operazioni

```
a = 0

while a < 5:
    print a
    a += 1

print "Finito!"
```

```
0
1
2
3
4
Finito!
```

Si può fare anche in console!

```
>>> a = 0
>>> while a != 6:
...     print a
...     a += 2
...
0
2
4
```

“...” indica che l'interprete Python ha rilevato un blocco, si aspetta delle istruzioni per riempirlo;

Ripetere le operazioni

```
for i in range(0,5):  
    print i  
  
stringa = "pippo"  
for s in stringa:  
    print s
```

```
0  
1  
2  
3  
4  
p  
i  
p  
p  
o
```

- Stampare i numeri dispari fino a 100;
- Stampare i tutti i multipli di 3 che sono pari, sfruttando l'**operatore modulo**, che dà il resto di una divisione:

Esercizi con i cicli

- Stampare i numeri dispari fino a 100;
- Stampare i tutti i multipli di 3 che sono pari, sfruttando l'**operatore modulo**, che dà il resto di una divisione:

```
print 3 % 2
1
print 10 % 4
2
print 10 % 2
0
```

- Lunghezza di una stringa:

```
>>> len("tarapia tapioco")
15
>>> stringa = "Questa stringa e' lunga 36 caratteri"
>>> len(stringa)
36
```

- Formattare le maiuscole

```
>>> print "berlino".capitalize()
Berlino
>>> stringa = "cambiare in maiuscolo"
>>> print stringa.upper()
CAMBIARE IN MAIUSCOLO
>>> stringa = "aeiou AEIOU"
>>> print stringa.lower()
aeiou aeiou
```

Trova e sostituisci

- Ricercare quante volte un termine compare all'interno di una stringa:

```
>>> "trentatre trentini entrarono a trento".count('tre')
4
>>> "trentatre trentini entrarono a trento".count('quattro')
0
```

- Cerca la posizione di una stringa all'interno di un'altra

```
>>> print "1111111111!1111111111111111".find("!")
11
```

- Sostituisce una stringa con un'altra

```
>>> print "sopra la panca la capra campa, sotto la panca la capra crepa".replace("
    ↪ capra", "pecora")
sopra la panca la pecora campa, sotto la panca la pecora crepa
>>> print "sa sa prova prova sa sa prova".replace("prova", "test", 2)
sa sa test test sa sa prova
```

Leggere stringhe dal terminale e...

- Stampare quanto sono lunghe
- Stamparle tutte maiuscole
- Stampare solo quelle che iniziano per a, b oppure c
- Terminare il programma quando scriviamo exit
- Rimpiazzare una parolaccia con #####

Grazie per l'attenzione!

GOLEM - Gruppo Operativo Linux Empoli



GOLEM – Gruppo Operativo Linux Empoli
presso “La Vela – Margherita Hack”
via Magolo, 32 – 50053 Empoli (FI)
tutti i martedì sera dalle 21.30 alle 24.00



Crediti & licenza

Questa serata è offerta da giuliof (giulio@glgprograms.it)
a partire dal materiale di Fiore (FabLab Toscana) e con contributi di lucam
utilizzando un template \LaTeX a cura di giomba.

Materiale rilasciato sotto GPL3 presso golem.linux.it