



# Python programming

programmazione  
in Python per il Liceo  
delle Scienze Applicate



Giulio Angiani  
I.I.S. "Blaise Pascal" - Reggio Emilia



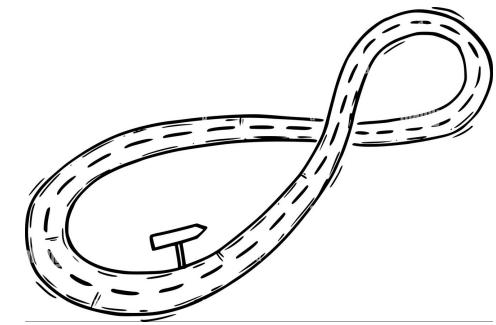
# Programmazione in Python



# Programmazione in Python

RECAP: Nella scorsa lezione abbiamo visto

- Cos'è un ciclo a numerosità predefinita
- L'istruzione **for**
- struttura base di una applicazione ciclica
  - istruzioni pre-ciclo
  - istruzioni in-ciclo
  - istruzioni post-ciclo
  - conteggio del numero di iterazioni effettuate



# Python - Ripetizione di operazioni

## ESERCIZIO A NUMEROSITÀ NON DEFINITA

- Leggere una sequenza di 5 numeri interi e contare quanti di essi sono pari

5 minuti per risolverlo....



# Python - Ripetizione di operazioni

## ESERCIZIO A NUMEROSITÀ NON DEFINITA

- Leggere una sequenza di 5 numeri interi e contare quanti di essi sono pari

```
# PRE-CICLO
contapari = 0 # inizializzazione della variabile che conterrà quanti pari verranno inseriti

for i in range(5):
    # NEL CICLO
    N = int(input("Inserisci un numero intero : ")) # acquisizione
    if N % 2 == 0: # se il resto della divisione per 2 è zero...
        contapari += 1

# POST-CICLO
print(f"Hai inserito {contapari} numeri pari") # stampa del risultato finale
```

PYTHON

```
Inserisci un numero intero : 4
Inserisci un numero intero : 6
Inserisci un numero intero : 5
Inserisci un numero intero : 3
Inserisci un numero intero : 2
Hai inserito 3 numeri pari
```

OUTPUT



# Python - Ripetizione di operazioni

## ESERCIZIO A NUMEROSITÀ NON DEFINITA

- Leggere la sequenza dei voti di informatica e stampare la media di quelli sufficienti... ([numerosità non definita](#))
- terminare la sequenza con il valore 0

[5 minuti per risolverlo....](#)

- **suggerimento:** nel ciclo dobbiamo fare varie cose...
- controllare se il voto inserito è sufficiente o no
- contare quante sono le sufficienze
- controllare se il valore inserito è pari a 0 e in questo caso terminare il ciclo



# Python - Ripetizione di operazioni

## ESERCIZIO A NUMEROSITÀ NON DEFINITA

- Leggere la sequenza dei voti di informatica e stampare la media di quelli sufficienti... (**numerosità non definita**)
- terminare la sequenza con il valore 0

```
# PRE-CICLO
numerosufficienze = 0    # inizializzazione della variabile che conterà quante sufficienze verranno inserite
sommasufficienze = 0      # inizializzazione della variabile che sommerà i valori delle sufficienze inserite

while True:      # NEL CICLO
    N = float(input("Inserisci la valutazione : "))  # acquisizione
    if N == 0: # se il valore inserito è zero termino la sequenza
        print("  >> Hai inserito 0 => esco dal ciclo....")
        break

    if N >= 6: # se il voto inserito è sufficiente
        print("  >> questo voto è sufficiente => conto e sommo....")
        numerosufficienze += 1      # conto una sufficienza in più
        sommasufficienze = sommasufficienze + N # sommo il valore alla somma attuale
    else:
        print("  >> questo voto NON è sufficiente => non faccio nulla....")

# POST-CICLO : calcolo il valore medio dividendo somma e numero
mediasufficienze = sommasufficienze/numerosufficienze
print(f"Hai inserito {numerosufficienze} sufficienze ") # stampa del risultato finale
print(f"La media delle sufficienze è {mediasufficienze}") # stampa del risultato finale
```

PYTHON



# Python - Ripetizione di operazioni

## ESERCIZIO A NUMEROSITÀ NON DEFINITA

- Leggere la sequenza dei voti di informatica e stampare la media di quelli sufficienti... (**numerosità non definita**)
- terminare la sequenza con il valore 0

```
Inserisci la valutazione : 3  
    >> questo voto NON è sufficiente => non faccio nulla....  
Inserisci la valutazione : 4  
    >> questo voto NON è sufficiente => non faccio nulla....  
Inserisci la valutazione : 8  
    >> questo voto è sufficiente => conto e sommo....  
Inserisci la valutazione : 7  
    >> questo voto è sufficiente => conto e sommo....  
Inserisci la valutazione : 9  
    >> questo voto è sufficiente => conto e sommo....  
Inserisci la valutazione : 5  
    >> questo voto NON è sufficiente => non faccio nulla....  
Inserisci la valutazione : 0  
    >> Hai inserito 0 => esco dal ciclo....  
Hai inserito 3 sufficienze  
La media delle sufficienze è 8.0
```

OUTPUT

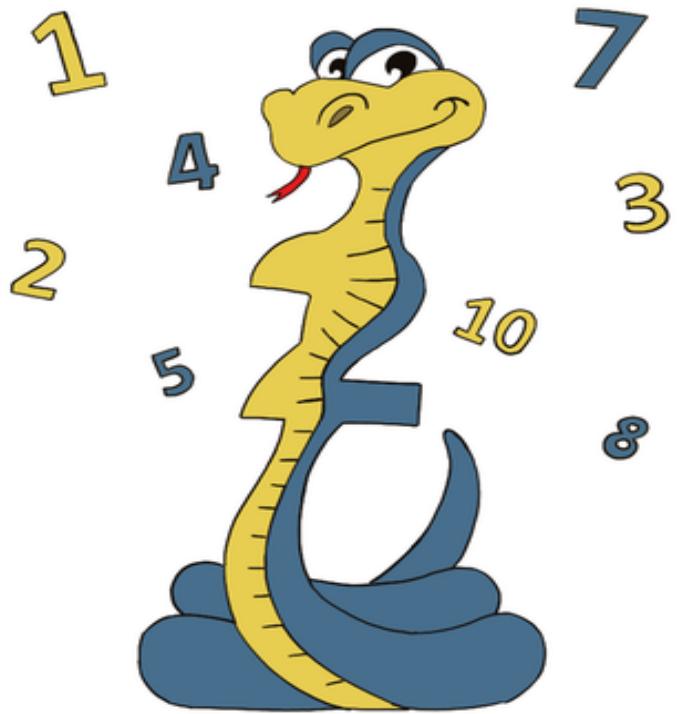




# Programmazione in Python

il nostro primo gioco...





# guess the number ?

@clcsimon | teachwithict.com

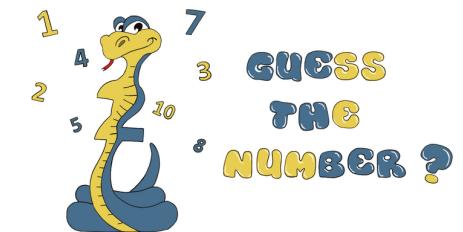
See also <https://www.teachwithict.com/numberguess.html>



# Python - Guess the number

## REGOLE DEL GIOCO

- Il computer sceglie un numero N a caso da 1 a 100
- L'utente inserisce un numero X da 1 a 100 e cerca di indovinare
- Se indovina, il programma risponde con "Bravo! Hai indovinato in T tentativi " (dove T è il numero di prove fatte dall'utente)
- Se non indovina, il programma risponde indicando se il numero da lui pensato è maggiore o minore di quello inserito dall'utente



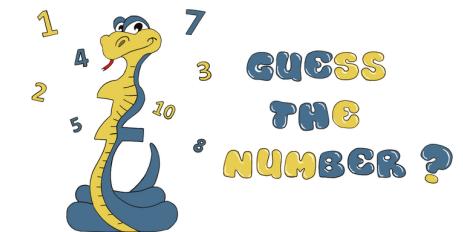
@clcsimon | teachwithict.com



# Python - Guess the number

Analizziamo il quesito : Dobbiamo risolvere 3 sotto problemi

- A) **Scegliere** un numero casuale N compreso fra 1 e 100
- B) **Confrontare** il numero X indicato dall'utente con quello N scelto dal computer
- C) **Contare** il numero di tentativi T effettuati dall'utente



@clcsimon | teachwithict.com

DOMANDE:

- Quante volte facciamo l'operazione A ?
- Quante volte facciamo l'operazione B ?
- Quante volte facciamo l'operazione C ?



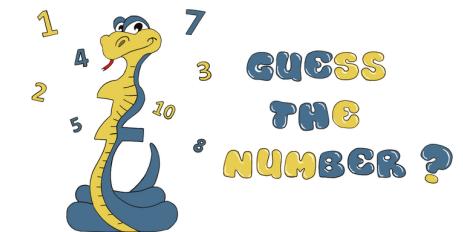
# Python - Guess the number

Analizziamo il quesito : Dobbiamo risolvere 3 sotto problemi

- A) Scegliere un numero casuale N compreso fra 1 e 100
- B) Confrontare il numero X indicato dall'utente con quello N scelto dal computer
- C) Contare il numero di tentativi T effettuati dall'utente

DOMANDE:

- Quante volte facciamo l'operazione A ?  $\Rightarrow$  1 sola volta
- Quante volte facciamo l'operazione B ?
- Quante volte facciamo l'operazione C ?



@clcsimon | teachwithict.com



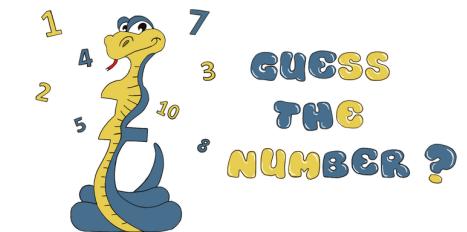
# Python - Guess the number

Analizziamo il quesito : Dobbiamo risolvere 3 sotto problemi

- A) Scegliere un numero casuale N compreso fra 1 e 100
- B) Confrontare il numero X indicato dall'utente con quello N scelto dal computer
- C) Contare il numero di tentativi T effettuati dall'utente

DOMANDE:

- Quante volte facciamo l'operazione A ?  $\Rightarrow$  1 sola volta
- Quante volte facciamo l'operazione B ?  $\Rightarrow$  finché N è diverso da X
- Quante volte facciamo l'operazione C ?



@clcsimon | teachwithict.com



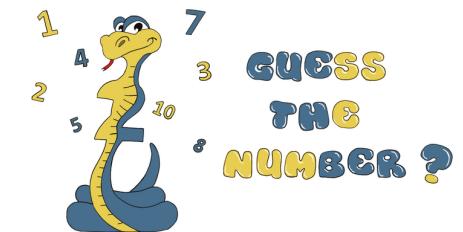
# Python - Guess the number

Analizziamo il quesito : Dobbiamo risolvere 3 sotto problemi

- A) Scegliere un numero casuale N compreso fra 1 e 100
- B) Confrontare il numero X indicato dall'utente con quello N scelto dal computer
- C) **Contare il numero di tentativi T effettuati dall'utente**

DOMANDE:

- Quante volte facciamo l'operazione A ?  $\Rightarrow$  1 sola volta
- Quante volte facciamo l'operazione B ?  $\Rightarrow$  finché N è diverso da X
- **Quante volte facciamo l'operazione C ?  $\Rightarrow$  Ogni volta che l'utente inserisce un numero**



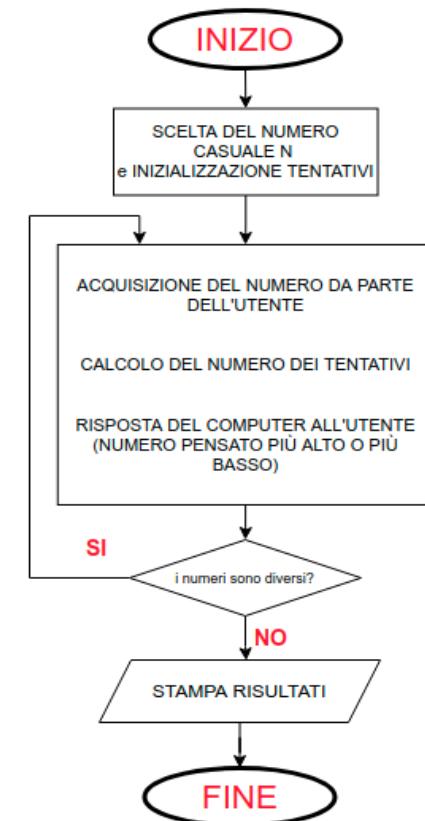
@clcsimon | teachwithict.com



# Python - Guess the number

Scriviamo un DAB con macroblocchi che raffiniamo volta per volta...

- FACCIAMO SCEGLIERE UN NUMERO CASUALE AL COMPUTER PRIMA DI ENTRARE NEL CICLO e INIZIALIZZIAMO IL NUMERO DI TENTATIVO (a zero)
- ENTRIAMO IN UN CICLO INDEFINITO CHE TERMINERÀ QUANDO L'UTENTE AVRÀ INSERITO IL NUMERO CORRETTO
- DOPO LA FINE DEL CICLO COMUNICHiamo I RISULTATI



# Python - Guess the number

## SCEGLIERE UN NUMERO CASUALE IN PYTHON

La libreria **random** offre le funzioni per i numeri casuali

Il metodo **randrange(N)** estrae un numero intero casuale fra 0 e N-1

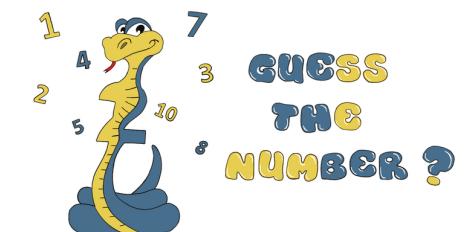
- per selezionare un numero a caso fra 1 ed N basta aggiungere 1....

```
# estrazione di un numero casuale fra 1 e 10
import random
n = random.randrange(10) + 1
print(n)
```

PYTHON

OUTPUT

5



@clcsimon | teachwithict.com



# Python - Guess the number

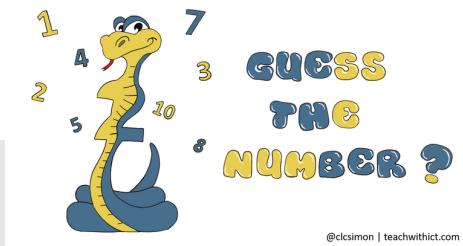
20 minuti per scrivere il nostro gioco....

Buon lavoro!

```
# PRE-CICLO
# estrazione di un numero casuale fra 1 e 100
import random
numero_da_indotinare = random.randrange(10) + 1
tentativi = ????

# IN-CICLO
while ????

# POST-CICLO
print(f"Hai indovinato in {tentativi} tentativi!")
```



@clcsimon | teachwithict.com



# Python - Guess the number

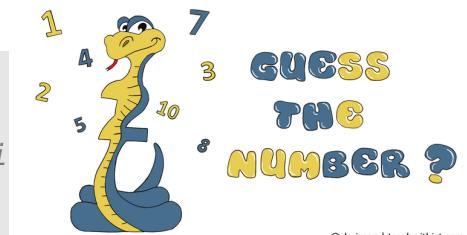
## La soluzione

```
import random # importo la libreria per i numeri casuali
N = random.randrange(1, 101) # scelgo un numero casuale da 1 a 100 compresi
tentativi = 0 # all'inizio metto a zero la variabile che conterrà il numero dei tentativi
print("Ho pensato un numero compreso fra 1 e 100! Prova ad indovinarlo....")

while True:
    # NEL CICLO
    X = int(input("Inserisci il tuo numero... : ")) # acquisizione
    tentativi += 1 # incremento i tentativi di 1
    if X == N: # se il valore inserito è uguale a quello "pensato" dal PC ho finito
        break

    if N > X: # se il valore scelto dal PC è maggiore di quello inserito
        print("  >> Il mio numero è più alto, riprova...")
    if N < X: # se il valore scelto dal PC è minore di quello inserito
        print("  >> Il mio numero è più basso, prova ancora...")

# POST-CICLO
print(f"Super! Hai indovinato in {tentativi} tentativi") # stampa del risultato finale
```



@clcsimon | teachwithict.com



# Un altro gioco... Ma solo virtuale!

I dadi!



# Un altro gioco... Ma solo virtuale!

Costruiamo un altro gioco usando i numeri casuali...

- All'inizio del gioco si assegnano 100 pascaleuro al giocatore.
- Ad ogni turno il giocatore sceglie
  - di puntare  $P$  pascaleuro (non può puntare più di quanto ha in cassa)
  - se puntare sui numeri bassi (da 2 a 6) o alti (da 8 a 12) oppure sul 7
  - di uscire dal gioco puntando 0
- Si simula il lancio dei dadi
- Se esce un numero compreso nella scelta del giocatore, il valore in cassa aumenta di  $P$  pascaleuro altrimenti diminuisce della stessa quantità
- Se esce il 7 ed il giocatore ha puntato sul 7, il valore in cassa aumenta di  $2P$  pascaleuro altrimenti diminuisce della stessa quantità
- Il gioco termina quando l'utente punta 0 o ha 0 pascaleuro in cassa



# Python - Guess the number

## domande

- Quante volte devo assegnare i 100 pascaleuro al giocatore?
- Quante volte devo simulare il lancio dei dadi ?
  - (suggerimento: lanciare 2 dadi da 6 non è come lanciare uno da 12...)
- Cosa devo fare dopo ogni lancio ?
- Quanto dura il ciclo ? Lo so dall'inizio?
- Devo controllare che la puntata P non superi il totale in cassa?
- Come chiedo all'utente di scegliere quanto puntare?
- Come chiedo all'utente di scegliere su cosa puntare?



**30 minuti per scrivere il nostro gioco....**

Buon lavoro!



# Python - Guess the number (soluzione)

```
import random
cassa = 100
while cassa > 0:
    print(f"Hai in cassa {cassa} pascaleuro")
    P = int(input("Quanto vuoi puntare... : ")) # acquisizione
    if P == 0:
        print("Hai terminato il gioco")
        break
    if P > cassa: # se il valore inserito è uguale a quello "pensato" dal PC ho finito
        print("Mi dispiace! Non puoi puntare più di quanto hai in cassa")
    else: # simulo il lancio dei dadi
        dado1 = random.randrange(1, 7)
        dado2 = random.randrange(1, 7)
        totale = dado1+dado2
        print("Inserisci A per numeri 8-12")
        print("Inserisci B per numeri 2-6")
        print("Inserisci S per puntare sul numero 7")
        tipopuntata = input("Su cosa vuoi puntare (A,B,S) ? : ") # acquisizione
        print("E' uscito il numero ", totale)
        if tipopuntata == "A" and totale < 7: # OK
            cassa = cassa + P
        elif tipopuntata == "B" and totale > 7: # OK
            cassa = cassa + P
        elif tipopuntata == "S" and totale == 7: # OK
            cassa = cassa + 2*P
        else: # non ho indovinato
            cassa = cassa - P
    print(f"Hai terminato! ti sono rimasti {cassa} pascaleuro ") # stampa del risultato finale
```

PYTHON



# Un esercizio di statistica!

Ma saranno veramente **casuali** ?

prova a simulare 10000 estrazioni da 1 a 10 e confronta quante volte esce l'1, il 2 etc...

Se sono veramente casuali dovrebbero essere circa 1000 per ogni numero...

====

Prova adesso a simulare 10000 lanci di 2 dadi...

Quali sono le distribuzioni statistiche dei valori?

Sono tutti equiprobabili?

**Buon lavoro**



# 10000 estrazioni da 1 a 10

Un codice bruttissimo ma funzionante...

```
import random
n1 = n2 = n3 = n4 = n5 = n6 = n7 = n8 = n9 = n10 = 0
for i in range(10000):
    N = random.randrange(1, 11)
    if N == 1: n1 += 1
    if N == 2: n2 += 1
    if N == 3: n3 += 1
    if N == 4: n4 += 1
    if N == 5: n5 += 1
    if N == 6: n6 += 1
    if N == 7: n7 += 1
    if N == 8: n8 += 1
    if N == 9: n9 += 1
    if N == 10: n10 += 1
print(f"Il numero 1 è uscito {n1} volte")
print(f"Il numero 2 è uscito {n2} volte")
print(f"Il numero 3 è uscito {n3} volte")
print(f"Il numero 4 è uscito {n4} volte")
print(f"Il numero 5 è uscito {n5} volte")
print(f"Il numero 6 è uscito {n6} volte")
print(f"Il numero 7 è uscito {n7} volte")
print(f"Il numero 8 è uscito {n8} volte")
print(f"Il numero 9 è uscito {n9} volte")
print(f"Il numero 10 è uscito {n10} volte")
```



# 10000 estrazioni da 1 a 10

Un codice bruttissimo ma funzionante...

Il numero 1 è uscito 999 volte  
Il numero 2 è uscito 1081 volte  
Il numero 3 è uscito 965 volte  
Il numero 4 è uscito 1005 volte  
Il numero 5 è uscito 1050 volte  
Il numero 6 è uscito 948 volte  
Il numero 7 è uscito 1022 volte  
Il numero 8 è uscito 1005 volte  
Il numero 9 è uscito 961 volte  
Il numero 10 è uscito 964 volte



Sono abbastanza equiprobabili....



# 10000 lanci di 2 dadi

```
import random
n2 = n3 = n4 = n5 = n6 = n7 = n8 = n9 = n10 = n11 = n12 = 0
for i in range(10000):
    d1 = random.randrange(1, 7)
    d2 = random.randrange(1, 7)
    N = d1+d2 # sommo i due dadi
    if N == 2: n2 += 1
    if N == 3: n3 += 1
    if N == 4: n4 += 1
    if N == 5: n5 += 1
    if N == 6: n6 += 1
    if N == 7: n7 += 1
    if N == 8: n8 += 1
    if N == 9: n9 += 1
    if N == 10: n10 += 1
    if N == 11: n11 += 1
    if N == 12: n12 += 1
print(f"Il numero 2 è uscito {n2} volte")
print(f"Il numero 3 è uscito {n3} volte")
print(f"Il numero 4 è uscito {n4} volte")
print(f"Il numero 5 è uscito {n5} volte")
print(f"Il numero 6 è uscito {n6} volte")
print(f"Il numero 7 è uscito {n7} volte")
print(f"Il numero 8 è uscito {n8} volte")
print(f"Il numero 9 è uscito {n9} volte")
print(f"Il numero 10 è uscito {n10} volte")
print(f"Il numero 10 è uscito {n11} volte")
print(f"Il numero 10 è uscito {n12} volte")
```

PYTHON



# 10000 lanci di 2 dadi

Un codice bruttissimo ma funzionante...

Il numero 2 è uscito 258 volte  
Il numero 3 è uscito 534 volte  
Il numero 4 è uscito 882 volte  
Il numero 5 è uscito 1150 volte  
Il numero 6 è uscito 1415 volte  
Il numero 7 è uscito 1675 volte  
Il numero 8 è uscito 1377 volte  
Il numero 9 è uscito 1086 volte  
Il numero 10 è uscito 804 volte  
Il numero 10 è uscito 541 volte  
Il numero 10 è uscito 278 volte



Assolutamente **NON equiprobabili** ....

Per saperne di più, parlane con il docente di Matematica!





Giulio Angiani  
I.I.S. "Blaise Pascal" - Reggio Emilia