

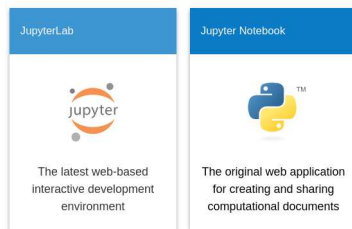
Incontro I

Lunedì 9 settembre

1 Proposta calendario prossimi incontri

data	ora inizio	ora fine
lunedì 9 settembre 2024	8:30	11:30
martedì 10 settembre 2024	8:00	11:00
lunedì 16 settembre 2024	14:00	16:00
giovedì 19 settembre 2024	14:00	16:00

2 Lavorare on line



jupyterLive



CoLab

3 Lavorare off line



3.1 Installazione di Python

Le modalità per l'installazione di Python variano a seconda del sistema operativo che si utilizza.

3.1.1 Python per Linux

Per installare Python su Linux (Ubuntu) è sufficiente aprire una shell e lanciare i seguenti comandi:

- `sudo apt-get install python3`
- `sudo apt-get install python-pip`
- `sudo apt-get install idle3`
- `sudo dnf install python3`
- `sudo dnf install python-pip`

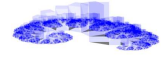
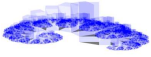
(Fedora)

- `sudo dnf install idle3`

3.1.2 Python per Windows e MacOS

Per quanto riguarda l'installazione di Python per i sistemi operativi Windows e MacOS, bisogna fare il download di un installer ed eseguirlo. In particolare, è sufficiente recarsi qui⁽¹⁾ e fare il download dell'installer (executable installer per Windows) a seconda del sistema operativo che si utilizza e della versione (se a 32 o a 64 bit). Durante l'installazione si consiglia di spuntare la casella "Add Python 3.7 to PATH" (questo vi verrà chiesto ad un certo punto dell'installazione).

¹se non funziona il link copia e incolla: <https://www.python.org/downloads/>



3.2 Installazione di Jupyter Lab

Per le seguenti installazioni abbiamo bisogno di aprire una shell (anche detto prompt, o console) dei comandi. Per aprire una shell sul sistema operativo Windows si può utilizzare la combinazione di tasti “Windows+R” poi digitare cmd ed infine premere invio.

Invece per aprire una shell sul sistema operativo MacOS è sufficiente cercare l’applicazione terminale e lanciarla. Installeremo Jupyter Lab usando il package manager pip di Python.

Per prima cosa aggiornate il package all’ultima sua versione, aprendo una shell ed eseguendo il comando

- `python3 -m pip install --upgrade pip`

Per installare Jupyter Lab adesso è sufficiente eseguire il seguente comando:

- `python3 -m pip install --user jupyterlab`

4 Comandi . . . per oggi, prima parte

4.1 Operazioni base

- somma: `+` (`3+4`)
- prodotto: `*` (`3*4`)
- divisione: `/` (`3/4`)
- potenza: `**` (`3**4`)

4.2 Librerie import

```
from sympy import *
init_printing()
from IPython.display import display, Math, Latex
```

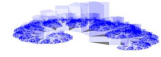
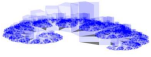
4.3 Cicli

4.3.1 For

```
a = 1
n = 20
for k in range(n) :
    a = Rational(a, (1+a))
    display(Math(latex(a)))
```

4.4 Scelte condizionate

- uguale: $a == b$ in \LaTeX $a = b$
- diverso: $a != b$ in \LaTeX $a \neq b$
- minore: $a < b$ in \LaTeX $a < b$
- minore o uguale: $a <= b$ in \LaTeX $a \leq b$
- maggiore: $a > b$ in \LaTeX $a > b$
- maggiore o uguale: $a >= b$ in \LaTeX $a \geq b$



4.4.1 if then else

prima una nuova libreria:

```
import random
```

```
N = random.randint(a, b)
```

Restituisce un intero N tale che $a \leq N \leq b$

```
X = random.random()
```

Restituisce un numero floating-point tale che $0.0 \leq X < 1.0$.

```
#esempio
a = random.randint(10, 400)
sta = str(a)
b = random.randint(10, 400)
stb = str(b)
if b > a:
    print(stb + " e' maggiore di "+sta)
else :
    print(stb + " e' minore o uguale di "+sta)
```

5 Esercizi ... prima parte

Problema 1. Genera un numero casuale $n \in \mathbb{N} : 18 \leq n \leq 59$ estrai un secondo numero $m \in \mathbb{N} : 18 \leq m \leq 59$
condizione 1

indica se $m > n$ o $m < n$ o $m = n$.

Stesso esercizio (condizione 1) ma ripeti fino a quando $|m - n| \leq 3$

Problema 2. Scrivi i primi venti numeri di Fibonacci.

I numeri di Fibonacci sono così "generati":

$$\begin{cases} a_0 = 0 \\ a_1 = 1 \\ a_{n+2} = a_n + a_{n+1} \quad n \in \mathbb{N} \end{cases}$$

Problema 3. In un contenitore ci sono 20 palline 15 rosa e 5 azzurre. Ne estraggo una.

- Come posso stimare la probabilità che esca una pallina rosa?
- Come posso stimare la probabilità che esca una pallina azzurra?

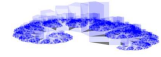
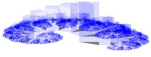
6 Comandi ... per oggi, seconda parte

6.1 Grafica

6.1.1 Libreria

```
import matplotlib.pyplot as plt
```

documentazione per matplotlib



6.1.2 Disegnare punti

```
Px = 2
Py = 1
figura = plt.figure(figsize=(5,5))
plt.plot(Px,Py,'bo')
Px = 1
Py = 2
plt.plot(Px,Py,'ro')
plt.show()
```

Il comando `random.random()` produce un numero pseudocasuale compreso nell'intervallo $[0, 1)$

7 Esercizi ... seconda parte

Problema 4. Disegna 100 punti di coordinate casuali contenuti nel rettangolo di diagonale AB dove $A = (-2, -1)$ e $B = (2, 1)$.

I colori dei punti sono rosso e blu scelti casualmente.

Problema 5. Nel quadrato di diagonale OA $O = (0, 0)$ $A = (1, 1)$ disegna n punti, colora i punti di rosso se si trovano all'interno del quarto di circonferenza di centro l'origine e raggio uno, blu altrimenti.

La circonferenza di centro l'origine e raggio 1 ha equazione $x^2 + y^2 = 1$, la distanza d_{AB} fra due punti $A = (x_1, y_1)$ e $B = (x_2, y_2)$ è data dalla formula:

$$d_{AB} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Problema 6. Con la tecnica usata nel problema 5 disegna il cerchio intero. Quadrato di lato 2, cerchio di centro l'origine e raggio 1.

Scrivi, poi, il quadruplo del rapporto fra il numero dei punti rossi e il totale n dei punti disegnati.

Problema 7. Disegna per punti il grafico della parabola di equazione $y = ax^2 + bx + c$. Imposta a, b, c .

Problema 8. Disegna il grafico della parabola di equazione $y = ax^2 + bx + c$. Imposta a, b, c . Nel grafico il vertice della parabola deve trovarsi al centro rispetto a x

Problema 9. Considera il quadrato $ABCD$ di lato $AB = 1$ inscrivi in esso un arco parabolico \mathcal{P}_{AB} di estremi A e B e vertice nel punto medio di CD .

Conviene utilizzare la parabola \mathcal{P} di equazione $y = 1 - 4x^2$, inscritta nel quadrato di lato AB , dove $A = \left(-\frac{1}{2}, 0\right)$ e $B = \left(\frac{1}{2}, 0\right)$.

Stima l'area del segmento parabolico.

Problema 10. Se si lancia una moneta 2 volte, la probabilità di ottenere una testa e una croce (in qualsiasi ordine) è pari al 50%. Se la moneta viene lanciata 4 volte, la probabilità di ottenere due teste e due croci, in qualsiasi ordine, è ancora pari al 50%? Costruisci un programma che ti permetta di rispondere alla domanda.

Problema 11. Zig-modo è piatto (2 dimensioni) per noi che lo vediamo da "fuori" per i zig-abitanti è il mondo.

Il generico zig-abitante si muove in modo un po' strano (per noi, naturale per lui): prima di fare un passo lancia il suo zig-dado (un quadrato con i lati colorati diciamo: colore 1, colore 2, colore 3, colore 4, se

- esce 1 avanti uno;
- esce 2 indietro uno;
- esce 3 ruota di 90° e avanti uno;
- esce 4 ruota di -90° e avanti uno

Costruisci una zig-passeggiata

Problema 12. Dato un triangolo equilatero ABC sia P_0 un suo generico punto interno si scelga a caso uno dei tre vertici del triangolo (ad esempio A), sia P_1 il punto medio del segmento P_0A . Il punto P_1 , ora, assume il ruolo di P_0 nella costruzione precedente, perciò si sceglie a caso un vertice del triangolo e si costruisce P_2 .

Ripetere la costruzione per n volte disegnando i punti così ottenuti.