

Graficare funzioni assegnate usando Python

Si realizzi uno script Python che permetta di visualizzare simultaneamente nell'intervallo $[-1,1]$ i grafici delle funzioni

$$y_1 = e^x \quad y_2 = \arccos(x) \quad y_3 = \tan(x) + \frac{1}{2}$$

soddisfacendo i seguenti requisiti estetici e funzionali:

- discretizzazione del dominio con almeno 1000 punti
- distinzione delle curve tramite colori e stili di linea differenti
 - y_1 : linea continua rossa
 - y_2 : linea tratteggiata verde
 - y_3 : linea tratto-punto nera
- dettagli: notazioni per gli assi, titolo del grafico, legenda descrittiva, griglia di riferimento

1) CODICE CHE RISOLVE L'ESERCIZIO

In accordo con le regole di sintassi di Python, le righe di codice che iniziano con # contengono esclusivamente commenti e vengono pertanto ignorate ai fini dell'esecuzione dell'algoritmo.

```
# Importiamo tutte le funzioni di NumPy
from numpy import *
# Importiamo Matplotlib per i grafici e assegniamogli un alias per brevità
import matplotlib.pyplot as plt
```

```
# Creiamo un array di 1000 valori x che vanno da -1 a 1
x = linspace(-1, 1, 1000)
```

```
# Definiamo le tre funzioni da graficare
y1 = exp(x)
y2 = arccos(x)
y3 = tan(x) + 0.5
```

```
# Creiamo il grafico delle tre funzioni
# Funzione y1 con linea rossa continua
plt.plot(x, y1, color='red', linestyle='-', label='Funzione $y_1$')
# Funzione y2 con linea verde tratteggiata
plt.plot(x, y2, color='green', linestyle='--', label='Funzione $y_2$')
# Funzione y3 con linea nera tratto-punto
plt.plot(x, y3, color='black', linestyle='-.', label='Funzione $y_3$')
```

```
# Aggiungiamo il titolo e le etichette agli assi
# Titolo del grafico
plt.title('Grafico delle funzioni $y_1$, $y_2$ e $y_3$')
# Etichetta dell'asse x
plt.xlabel('x')
# Etichetta dell'asse y
plt.ylabel('y')

# Aggiungiamo la legenda per distinguere le funzioni
plt.legend()

# Mostriamo la griglia nel grafico per facilitarne la lettura
plt.grid(True)

# Visualizziamo il grafico
plt.show()
```

2) ULTERIORI SPIEGAZIONI E COMMENTI

- $x = \text{linspace}(-1, 1, 1000)$

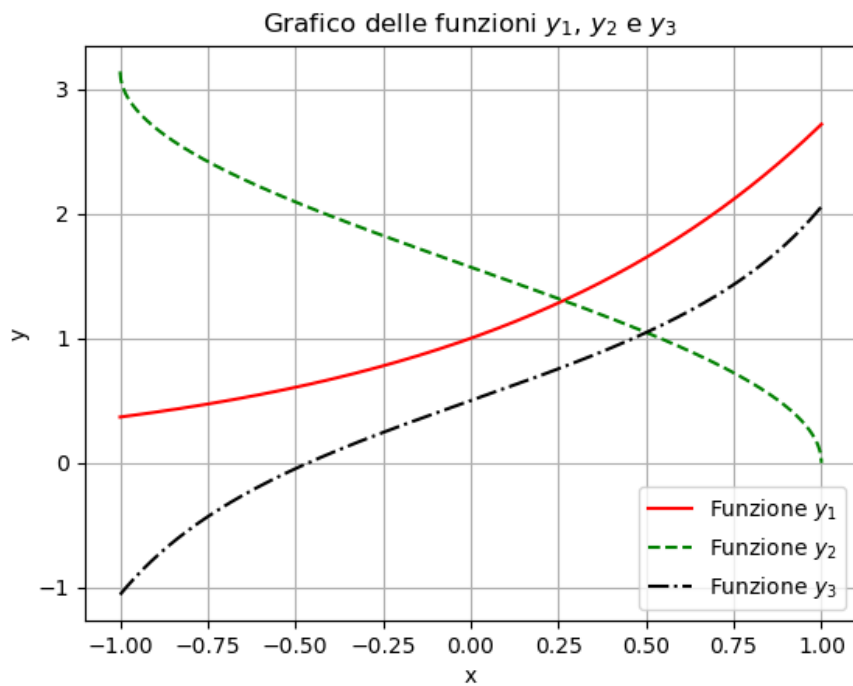
La funzione $\text{linspace}(a, b, c)$ crea un array di c valori equidistanti tra a e b . Questo intervallo sarà utilizzato per calcolare le immagini delle tre funzioni

- $\text{plt.plot}(\dots)$

Ogni riga di questa funzione crea una curva nel grafico, associando il valore di x con il valore della funzione y_i . Ogni curva ha un colore e uno stile di linea specifico, oltre a un'etichetta (che comparirà nella legenda)

3) OUTPUT

Di seguito è riportato il grafico richiesto, ottenibile facendo eseguire il codice sopra da un qualsiasi ambiente di sviluppo per Python, come ad esempio Spyder.



A cura di Gabriele Leggeri