Compito del 11/09/2018 da svolgere in Python

[1] Costruire la matrice quadrata V di ordine n con elementi:

$$v_{ij} = x_i^{n-j},$$

$i = 1, ..., n$, e $j = 1, ..., n$ al variare di $n = 3, ..., 10$ e $x = (x_0, \cdots, x_n)$ vettore di punti equispaziati in $[1, 2]$.
Calcolare il numero di condizionamento $K(A) = ||A||||A^{-1}||$ al variare di n in norma infinito (se viene complicato calcolare l'inversa di V, usare la libreria *linalg* con *from munpy.linalg import inv* e calcolare *inv(V)*).
Infine plottare tale numero di condizionamento al variare di n in scala semi-logaritmica e commentare il risultato.
Ricordarsi di usare: *import matplotlib*, *import matplotlib.pyplot as plt*, e per plottare in scala semi-logaritmica usare *plt.semilogx(...)*, *plt.show(...)* [10 punti]

[2] Data la funzione

$$f(x) = \frac{1}{1 + \log x}, \quad x \in [0.1, 1.0].$$

Verificare se il polinomio interpolante costruito con il metodo di Newton, converge alla funzione f all'aumentare del numero di nodi equidistanti per $n = 10, 20, 40, 80$. Calcolare l'errore commesso per ogni n e fare il grafico di tale errore in funzione di n. [10 punti]

[3] Si applichi il metodo dei trapezi composito per la stima dell'integrale:

$$I := \int_0^1 \exp(x^2) dx,$$

considerando una suddivisione dell'intervallo di integrazione $[0; 1]$ in N sottointervalli di uguale ampiezza $H = 1/N$, con $N = 4, 8, 16, 32, 64$.

Inoltre si calcolino le quantità:

$$Err = \frac{E_N}{E_{2N}}, \quad \text{e} \quad \log_2(Err),$$

al variare di N, e si giustifichino i risultati ottenuti.

[10 punti]