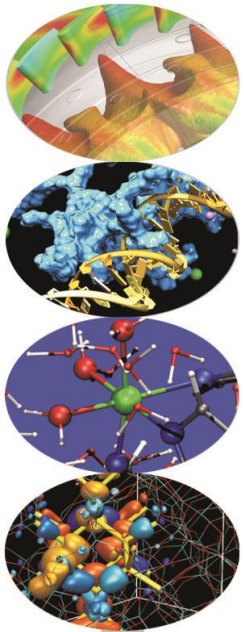
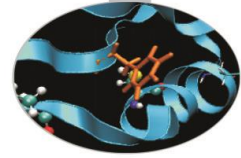




Esercitazione

Pylab



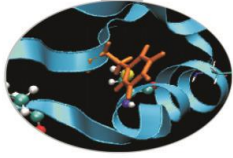


Esercitazione

Esercizio 0 (Sintassi pylab base)

- Crea un plot della funzione $\sin(x)$, usando una linea rossa in dashed style (--) aggiungi il titolo, il nome agli assi in verde e con dimensione 20, e una legenda
- Creare un grafico con due subplot (2 righe 1 colonna) sul primo grafico disegnare la funzione $\sin(x)$ e $\cos(x)$ sul range $[0, 2\pi]$, settare titolo e assi con un fontsize differente, aggiungere la legenda. Su secondo grafico disegnare la funzione $\sin(x)\cos(x)$ sul medesimo intervallo.
- Crea una sequenza di 10 immagini in cui la funzione $\sin(x)$ evolve nel tempo. Utilizza lo stesso plot e la funzione `draw()` per caricare la nuova immagine.

(Solution: `pylab_base.py`)



Esercitazione

Esercizio 1 (Sintassi pylab)

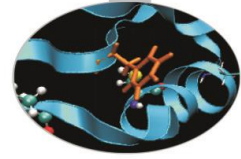
Leggere le concentrazioni di CO₂ sul monte Cimone. (usare opportunamente la funzione loadtxt). Fare un plot delle medie annuali (ultima colonna) rispetto all'anno.

Inserire nome degli assi e titolo.

Interpolare linearmente i dati (tramite la funzione numpy.polyfit) e rappresentare graficamente.

Rappresentare sullo stesso grafico l'andamento delle temperature nei diversi mesi.

(Solution: plot_mountCM.py)



Esercitazione

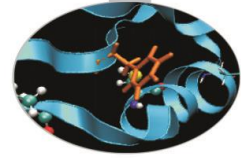
Esercizio2 (pylab):

La funzione $\sin(x)$ può essere approssimata con uno sviluppo polinomiale:

$$\sin x \approx S(x, n) = \sum_{j=0}^n (-1)^j \frac{x^{2j+1}}{(2j+1)!}$$

Visualizzare graficamente la qualità dell'approssimazione è per $n=8, 12, 16, 20$ per x in $[0, 4\pi]$ con la funzione subplot.

([RESULTS sin_appx.py](#))



Esercitazione

Esercizio3 (pylab):

Leggere gli eventi meteo per la città di Bloomington nel 2012. Caricare esclusivamente i campi timestamp, max temp, mean temp, min temp, events (le colonne 0,1,2,3,21), sfruttando opportunamente la funzione `numpy.loadtxt`.

Scrivere una funzione `temp_plot(dates, temps)` che salvi su file il grafico della temperatura rispetto ai giorni, e la sua interpolazione lineare (`numpy.polyfit`). Salvare i grafici della temperatura media, massima e minima.

Scrivere una funzione `hist_events(dates, events)` che calcoli e salvi su file l'istogramma in frequenza dei vari eventi ('Rain', 'Thunderstorm', 'Snow', 'Fog') sui vari mesi dell'anno.

(Solution: `plot_weather.py`)