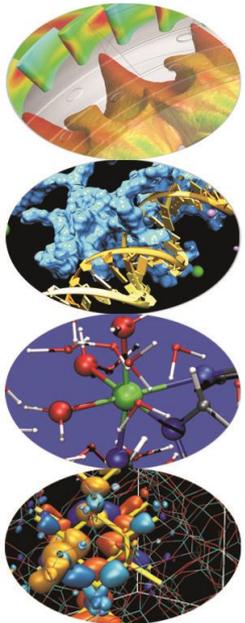
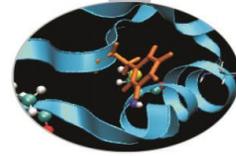


# Esercitazioni

Funzioni





# Esercitazioni

## Esercizio 1 (Sintassi funzioni):

Scrivere una funzione che calcoli il massimo e il minimo valore di una funzione  $f(x)$  definita su un intervallo  $[a,b]$  valutata in  $n$  (default =2000) punti su tale intervallo.

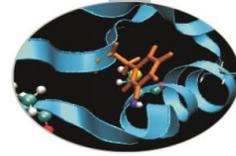
(Solution: max\_min\_f.py)

## Esercizio2 (Recursion):

Implementare una funzione per verificare se una stringa  $s$  è palindroma.

Implementare la funzione in due diverse versioni: la versione iterativa e una versione ricorsiva.

(Solution: palindroma.py)



# Esercitazioni

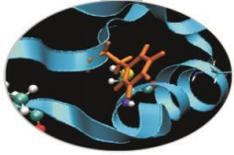
## Esercizio3 (Lambda function):

Implementare una funzione che prende in input una lista di parole e ritorna una lista con la lunghezza di ciascuna parola. Usando un ciclo `for` e successivamente la funzione `map`.

Implementare una funzione che presa una lista di parole ritorni una lista con le parole con lunghezza maggiore di 3. Usare la funzione `filter`  
(Solution: `wordFunc.py`)

## Esercizio4 (Lambda expression):

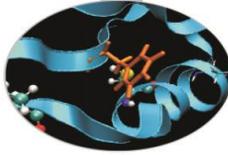
Implementare una funzione per il calcolo del prodotto scalare tra due vettori. Successivamente sfruttare opportunamente le funzioni `map`, `reduce` e le `lambda expression` e reimplementare tale prodotto.  
(Solution: `scalar_prod.py`)



# Esercitazioni

## Esercizio 5:

- Scrivere una funzione `path_length(L_vertex)` per il calcolo della lunghezza della traiettoria su un piano percorsa da un corpo in moto in N istanti di tempo.
- Si assuma di racchiudere le informazioni relative alle N coordinate spaziali del corpo in una lista `L_vertex = [[x1,y1],[x2,y2],...[xn,yn]]`



# Esercitazioni

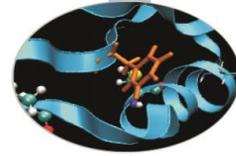
## Esercizio 6:

- Usare la funzione `path_length(L_vertex)` per calcolare la distanza percorsa tra i punti (1,1)-(3,2)-(2,3)-(1,1).
- Il valore  $\pi$  è equivalente alla circonferenza di un cerchio di raggio 0.5.
- Approssimare il calcolo di  $\pi$  discretizzando la circonferenza con una spezzata poligonale. Le  $N$  coordinate dei punti  $(x_i, y_i)$  [ $i=0, \dots, N-1$ ] della spezzata sono date da:

$$x_i = \frac{1}{2} \cos(2\pi i / N) \qquad y_i = \frac{1}{2} \sin(2\pi i / N)$$

- Utilizzando la funzione del punto precedente calcolare la lunghezza della spezzata e al variare di  $N$  l'accuratezza della soluzione.

([Solution: pi\\_appx.py](#))



# Esercitazione

## Esercizio 7:

- Implementare il metodo della bisezione per la ricerca degli zeri di una funzione  $f(x)$  continua definita su un intervallo  $[a,b]$ .
- Partendo dall'intervallo  $[a,b]$  si valutano i due sottointervalli  $[a,(a+b)/2]$  e  $[(a+b)/2, b]$ , se  $f(a)*f((a+b)/2)<0$  si reitera sul sottointervallo  $[a,(a+b)/2]$  e viceversa se  $f((a+b)/2)*f(b)<0$ . La procedura viene ripetuta fino al raggiungimento della tolleranza prescritta  $|b_i - a_i| < \epsilon$ .
- Applicarlo al calcolo degli zeri della funzione  $f(x)=x^3 - 10x^2+5$  nell'intervallo  $[0.6, 0.8]$ , definendo una funzione bisection con il seguente prototipo:

bisection(f,a,b,toll)  $\rightarrow$  x

(Solution: bisection.py)