

Application

Exercices d'application

Exercice 1 – Calcul de distance

Soient n points de coordonnées $P_1(x_1, y_1, z_1), \dots, P_n(x_n, y_n, z_n)$. On cherche à déterminer la distance entre chacun de ces points.

On donne la fonction `distance` permettant de déterminer la distance entre deux points.

```
def distance(p1, p2):
    d0 = p2[0]-p1[0]
    d1 = p2[1]-p1[1]
    d2 = p2[2]-p1[2]
    d = (d0**2+d1**2+d2**2)**(1/2)
    return d
```

Question 1 Préciser la signature de cette fonction sous forme de commentaires. On ajoutera aussi les annotations de type de cette fonction.

Question 2 Donner un (ou des) test(s) sous la forme d'assertions(`assert`) permettant de valider les entrées de la fonction.

Question 3 Proposer des tests permettant de vérifier les sorties de la fonction.

On donne maintenant la fonction suivante permettant de calculer la longueur d'un chemin constitué de n points.

```
def longueur(L:list)->float:
    """
    Déterminer la longueur du chemin L
    Entrée :
    - L:list : liste des points constitués de leurs coordonnées : [[x0,y0,z0],...[xn,yn,zn]]
    Sortie :
    - longueur du chemin
    """
    l = 0
    for i in range(len(L)):
        l = l+distance(L[i],L[i+1])
    return l
```

Soient $p_1 = [0,0,0]$, $p_2 = [1,0,0]$ et $p_3 = [2,0,0]$ trois points.

Question 4 Comment utiliser la fonction `longueur` pour déterminer la distance du chemin $p_1 - p_2 - p_3$?

Question 5 La fonction `longueur` fonctionne-t-elle pour ce chemin ? Si elle ne fonctionne pas, modifier la fonction.

Question 6 Donner un (ou des) test(s) sous la forme d'assertions(`assert`) permettant de valider les entrées de la fonction `longueur`.