

```

1  import numpy as np
2
3  f=open('JoueursTop14.txt','r')
4  liste_lignes=f.readlines()
5  f.close()
6
7  liste_joueurs=[]
8
9  for ligne in liste_lignes:
10     champs=ligne.split(';')
11     d={}
12     d['Poste']=champs[3]
13     d['Poids']=float(champs[4])
14     d['Taille']=float(champs[5])
15     liste_joueurs.append(d)
16
17  def postes():
18     D={} #le dictionnaire que l'on va créer
19     for d in liste_joueurs:
20         D[d['Poste']] =0 #on crée le couple (clef, valeur)
21         #si cette entrée existait déjà on réécrit par dessus, pas grave
22     return D
23
24  def liste_joueurs_distances(poids, taille):
25     liste_distances=[]
26     for d in liste_joueurs:
27         dl={} #le dictionnaire qu'on va ajouter à liste_distances
28         dl['Poste']=d['Poste']
29         dl['distance']=np.sqrt((poids-d['Poids'])**2+(taille-d['Taille'])**2)
30         liste_distances.append(dl)
31     return liste_distances
32
33
34  def KNN(k,poids,taille):
35     D=postes() #le dictionnaire des postes à mettre à jour
36     liste_distances=liste_joueurs_distances(poids,taille)
37     #on va trier cette liste par ordre croissant des distances obtenues:
38     liste_distances_triée=sorted(liste_distances,key=lambda d:d['distance'])
39
40     for i in range(k):#on va parcourir les k premiers éléments de la liste triée
41         d=liste_distances_triée[i]
42         D[d['Poste']] +=1
43     return(D)
44
45
46  def prediction(k,poids,taille):
47     D=KNN(k,poids,taille)
48     #on va chercher quel est le poste majoritaire parmi mes k voisins
49     nb_max=-np.inf
50     for clef,valeur in D.items():
51         if valeur>nb_max: #ce test va être vrai au premier tour de boucle
52             poste_prédit=clef
53             nb_max=valeur
54     return poste_prédit
55
56  print(prediction(8,100,190))
57  #Si je pèse 100kg et que je mesure 1m90, alors je jouerai CENTRE
58
59
60

```