

54 Lancer de poids

Programme à compléter

Cet exercice a pour but de comprendre le principe du calcul numérique d'un mouvement dont l'expression analytique n'est pas connue, en utilisant la deuxième loi de Newton dans une version discrète, comme cela a été fait en classe de première.

Fichiers Python

Programme à compléter
Fiche d'accompagnement

➔ hatier-clic.fr/pct336

Prérequis théoriques

- Deuxième loi de Newton

L'objectif de la question **a** est de **justifier les relations** permettant d'incrémenter les coordonnées des vecteurs vitesse et position.

La question **b** vise à **comprendre quelle condition** est choisie pour terminer la boucle.

Enfin, à la question **c**, il faut **compléter le programme** pour calculer la portée du tir.

Programme à compléter

```

1  from pylab import *
2
3  print("*****")
4  print("*   Lancer de poids   *")
5  print("* Exercice 54 p. 336 *")
6  print("*****Hatier 2020**")
7
8  ### NE PAS MODIFIER
9  m=4.0   #Masse du systeme (en kg)
10 g=9.81  #Norme du champ de pesanteur (en N/kg)
11 k=0.1   #Coefficient de frottement (en N.s/m)
12 Tx=-10  #Force due au vent horizontal (en N)
13 Dt=1E-2 #Pas de temps en seconde
14 v0=10   #Norme de la vitesse initiale au moment du lancer (en m/s)
15 h=2.2   #Altitude du la boule au moment du lancer (en m)
16 angle=40 #Angle de tir (en °)
17
18 # Initialisations
19 t=[0]
20 x=[0]
21 y=[h]
22 vx=[v0*cos(angle/180*pi)]
23 vy=[v0*sin(angle/180*pi)]
24 p=0
25
26 # BOUCLE calculant le mouvement
27 while y[p]>=0:
28     t.append(t[-1]+Dt)
29     #Coordonnees de la somme des forces
30     Fx=-k*vx[p]+Tx
31     Fy=-m*g-k*vy[p]
32
33 ##### A COMMENTER exercice 54 question a #####
34     #Coordonnees du vecteur vitesse a l'instant t + Dt
35     vx.append(vx[p]+Fx*Dt/m)
36     vy.append(vy[p]+Fy*Dt/m)
37     #Coordonnees du vecteur position a l'instant t + Dt
38     x.append(x[p]+vx[p]*Dt)
39     y.append(y[p]+vy[p]*Dt)
40
41 ##### NE PAS MODIFIER
42     p=p+1
43
44 ##### A COMPLETER exercice 54 question c #####
45 portee=... # Expression de la portee
46
47 print("Pour l'angle ",angle,"° la portee du tir est ",portee," m")
48
49 # Trace de la trajectoire du système
50 plot(x,y)
51 xlabel("x (en m)")
52 ylabel("y (en m)")
53 xlim(0,1.1*max(x))
54 ylim(0,1.1*max(y))
55 title("Trajectoire")
56 grid(True)
57 show()

```

Module importé

Le module `pylab` fournit les instructions graphiques utiles à l'exercice.

Ne pas modifier : données

Les données ne sont pas nécessairement à modifier pour répondre aux questions de l'exercice. On pourra faire ensuite varier ces différents paramètres pour étudier leur influence sur la trajectoire (mettre un vent très fort par exemple).

Ne pas modifier : initialisations

Les listes des dates, des positions et des vitesses sont initialisées avec leurs valeurs à $t = 0$ s.

Attention Les valeurs des fonctions `cos` et `sin` sont calculées en radians.

Ne pas modifier : initialisation de la boucle, calcul du temps et de la somme des forces

Les listes sont complétées avec l'instruction `append` : `L.append(X)` ajoute l'élément `X` à la fin de la liste `L`. La dernière valeur de la liste `L` est `L[-1]`. La condition de la boucle `while` doit être expliquée à la question **b**.

À expliquer : calculs des coordonnées

Ces égalités sont à justifier à la question **a** en faisant appel à la deuxième loi de Newton dans sa version discrète, vue en première, et à la définition de la vitesse.

À compléter : calcul de la portée

L'expression de la portée est à compléter à la question **c**. La fonction `max(L)` donne la valeur du maximum de la liste `L`.

Ne pas modifier : tracé de la trajectoire

La fin du programme réalise le tracé de la trajectoire.