

NOUVEAU PROGRAMME : DECRIRE UN MOUVEMENT

Cadre de l'activité

Notions et contenus abordés

Description du mouvement d'un système par celui d'un point.

Position. Trajectoire d'un point.

Capacités exigibles. Activités expérimentales support de la formation

Décrire le mouvement d'un système par celui d'un point et caractériser cette modélisation en termes de perte d'informations.

Caractériser différentes trajectoires.

Capacité numérique

Représenter les positions successives d'un système modélisé par un point lors d'une évolution unidimensionnelle ou bidimensionnelle à l'aide d'un langage de programmation.

I. But de la séance

Un centre de lancer de haches a ouvert récemment à Strasbourg. Le lien suivant vous permet de visualiser quelques [lancers](https://www.youtube.com/watch?time_continue=64&v=lbCpi_k7rF4) un peu particulier. [https://www.youtube.com/watch?time_continue=64&v=lbCpi_k7rF4]



Comment peut-on décrire qualitativement (puis quantitativement) la trajectoire de ces haches et d'autres objets plus généralement ?

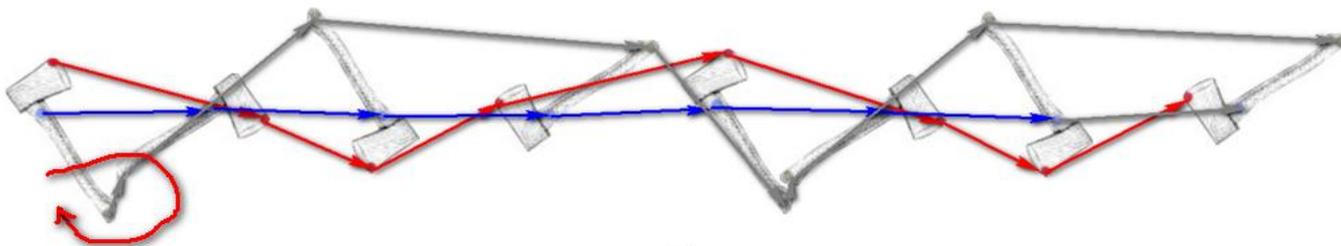
II. Mouvement d'un système. Mouvement du centre de gravité.

La hache tourne sur elle-même et se déplace plus ou moins horizontalement (hélicoïdal).

Trajectoire du point Rouge :



Trajectoire des points Rouges, bleus et gris :

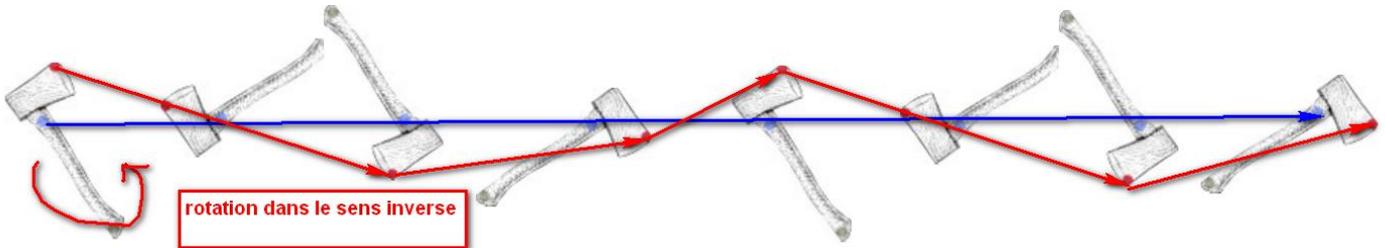


C'est la trajectoire du point « bleu » qui se rapproche le plus de la trajectoire de la hache

Le point bleu est le centre d'inertie de la hache

A retenir :

Dans de nombreux cas (et au moins en seconde) nous étudierons systématiquement la trajectoire de ce point particulier* plutôt que la trajectoire de l'objet lui-même. On appelle 'cinématique' la branche de la physique qui étudie le mouvement. On appelle 'système' l'objet dont on étudie la trajectoire. Enfin on appelle 'centre de gravité' le point particulier* dont le mouvement modélise celui du système.



La modélisation du système par un point se caractérise par une perte d'information sur le système.

Impossible de savoir dans quel sens tourne la hache par exemple...

III. Modéliser des trajectoires dans un langage de programmation (Python)

Comment peut-on simuler le déplacement d'un point dans le langage Python ?

Python permet de dessiner très rapidement de fenêtres graphiques dans lesquelles on peut placer des objets géométriques.

Programme pour tracer un point de coordonnées $x = 250$, $y = 20$:

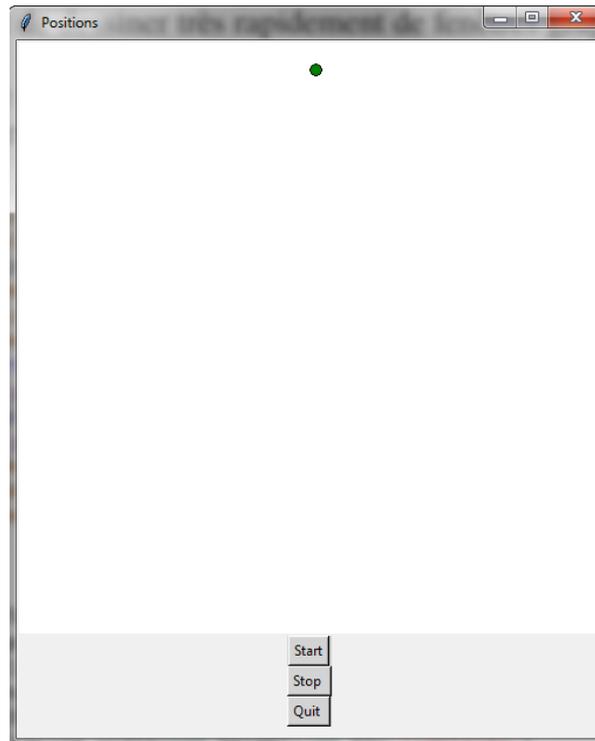
```
1  #!/usr/bin/python
2  # -*- coding: utf-8 -*-
3  # DLommele - lyceemangin
4  from tkinter import *
5  #####
6  # Fonctions #
7  #####
8  def Start():
9      pass
10 def Stop():
11     pass
12 #####
13 # PROGRAMME PRINCIPAL #
14 #####
15 t=0
16 x=250
17 y=20
18 racine=Tk()
19 racine.geometry("500x600")
20 racine.title("Positions")
21 fond=Canvas(racine, bg='white',width=500,height=500,bd=4)
22 fond.pack()
23 f=Frame(racine,width=500,height=100,bg='lightgrey')
24 f.pack()
25 debut=Button(f,text="Start",bg='lightgrey')
26 debut.pack()
27 fin=Button(f,text="Stop ",bg='lightgrey')
28 fin.pack()
29 quitter=Button(f,text="Quit ",bg='lightgrey', command=quit)
30 quitter.pack()
31 rond=fond.create_oval(x,y,x+10,y+10,fill='green')
32 racine.mainloop()
```

Coordonnées (x, y) du point

Taille (en pixel) de la fenêtre

Taille du point , couleur du point

Résultat du programme :



Le programme 2 permet d'afficher le point qui se déplace toutes les 500 ms avec des nouvelles coordonnées :

```
1 #!/usr/bin/python
2 # -*- coding: utf-8 -*-
3 # DLommele - lyceemangin
4 from tkinter import *
5 #####
6 # Fonctions #
7 #####
8 def Start():
9     global x,y,t
10
11     x=x+30
12
13     fond.coords(rond,x,y,x+10,y+10)
14
15     racine.after(500,Start)
16 def Stop():
17     pass
18 #####
19 # PROGRAMME PRINCIPAL #
20 #####
21 t=0
22 x=2
23 y=250
24 racine=Tk()
25 racine.geometry("500x600")
26 racine.title("Positions")
27 fond=Canvas(racine, bg='white',width=500,height=500,bd=4)
28 fond.pack()
29 f=Frame(racine,width=500,height=100,bg='lightgrey')
30 f.pack()
31 debut=Button(f,text="Start",bg='lightgrey',command=Start)
32 debut.pack()
33 fin=Button(f,text="Stop ",bg='lightgrey')
34 fin.pack()
35 quitter=Button(f,text="Quit ",bg='lightgrey', command=quit)
36 quitter.pack()
37 rond=fond.create_oval(x,y,x+10,y+10,fill='green')
38 racine.mainloop()
```

x varie, on ajoute 30 à chaque implémentation...

On fait varier x toutes les 500 ms

On lance le programme 3 qui laisse les points affichés à l'écran. Le point se déplace toutes les 0.5 s

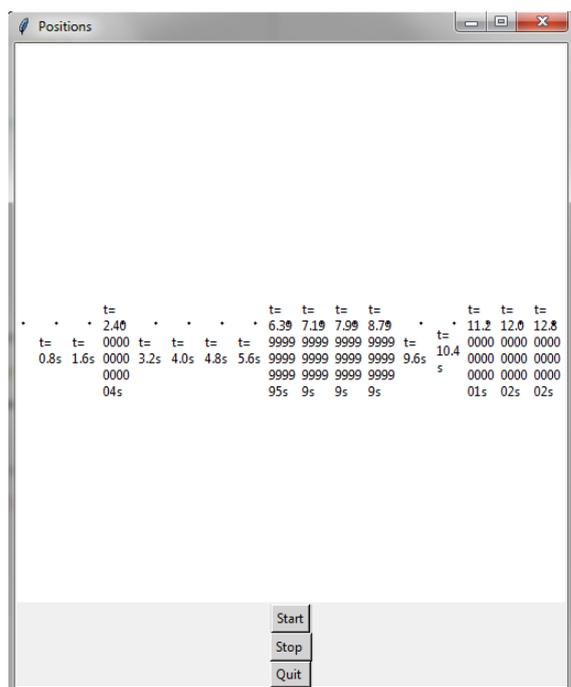
On modifie le programme n°3 pour que le rond rouge se déplace de la gauche vers la droite toutes les 800 ms :

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
# DLommele - Lyceemangin
from tkinter import *
#####
# Fonctions #
#####
def Start():
    global x,y,t
    fond.create_oval(x+4,y+4,x+6,y+6,fill='black')
    x=x+30
    t=t+0.8
    fond.coords(fond,x,y,x+10,y+10)
    a="t=\n"+str(t)+"="
    fond.create_text(x,y+20,text=a,width=25)
    racine.after(800,Start)
def Stop():
    pass
#####
# PROGRAMME PRINCIPAL #
#####
t=0
x=2
y=250
racine=Tk()
racine.geometry("500x600")
racine.title("Positions")
fond=Canvas(racine, bg='white',width=500,height=500,bd=4)
fond.pack()
f=Frame(racine,width=500,height=100,bg='lightgrey')
f.pack()
debut=Button(f,text="Start",bg='lightgrey',command=Start)
debut.pack()
fin=Button(f,text="Stop ",bg='lightgrey')
fin.pack()
quitter=Button(f,text="Quit ",bg='lightgrey', command=quit)
quitter.pack()
rond=fond.create_oval(x,y,x+10,y+10,fill='green')
racine.mainloop()
```

On affiche la valeur de t sous le point

On change t en 800 ms

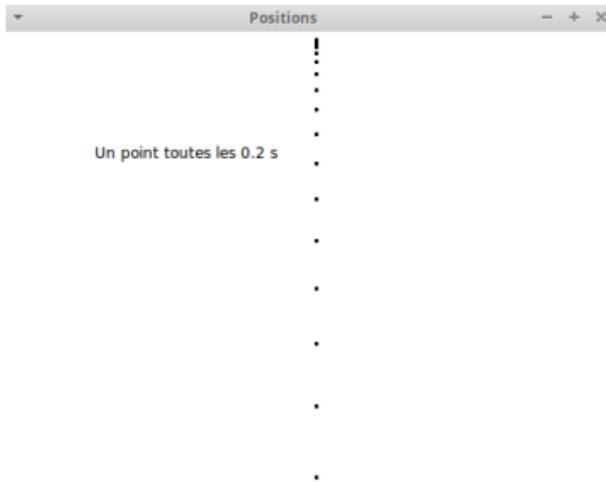
On exécute le programme :



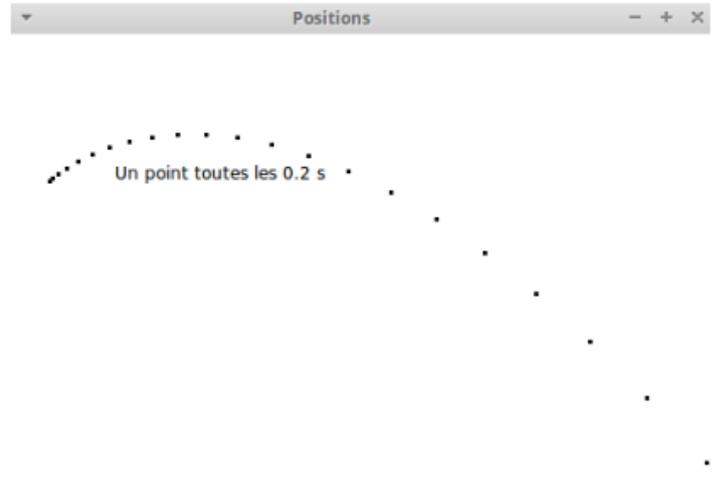
IV. Caractériser des trajectoires différentes

Si la trajectoire du point considéré dans l'étude du mouvement est une droite, on dit que le mouvement est rectiligne.
Si l'écart entre les points reste le même pour des intervalles de temps fixés, on dit que le mouvement est uniforme.

On lance les programmes n°4 à 7... :



Programme 4 : Mouvement Rectiligne Accéléré



Programme 5 : Mouvement curviligne accéléré



Programme 6 : Mouvement Rectiligne accéléré



Mouvement circulaire uniforme