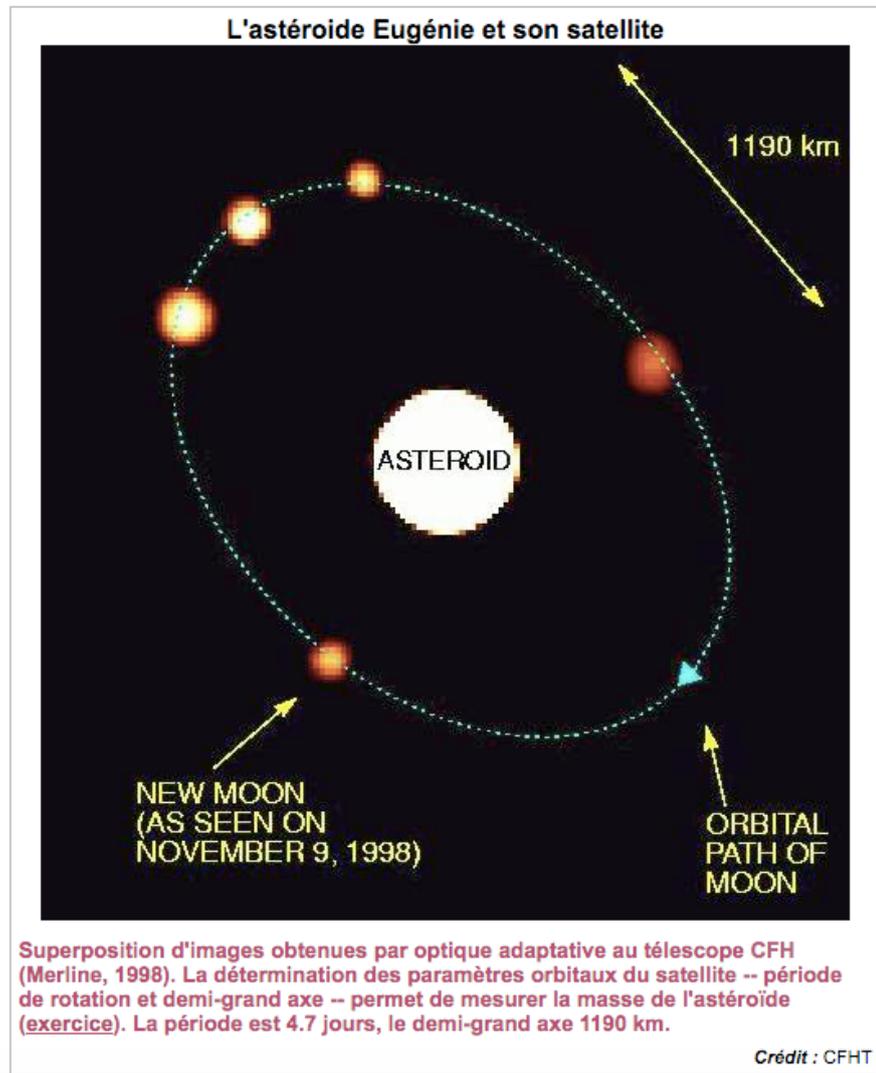




PROGRAMMER EN PYTHON POUR ALLER PLUS LOIN N°2 : LOI DE LA GRAVITATION

Le but est de continuer à se familiariser avec un langage de programmation à travers cette fois l'étude du mouvement d'un astéroïde. Cette fois encore, une photographie et un tableau de positions successives sont exploités dans le but de parvenir à une description quantitative d'un phénomène ayant trait à la mécanique. Cette fiche reprend les méthodes utilisées pour illustrer les capacités numériques du programme de la classe de seconde, en les adaptant au cas d'un mouvement d'astéroïde.

Le but est de déterminer la masse de l'astéroïde Eugénie à l'aide du mouvement de son satellite. Pour cela, la validité de la troisième loi de Kepler est d'abord testée en exploitant un tableau relatif aux orbites et période de révolution des planètes au tour du Soleil. Ces données sont récupérées librement sur le site de l'observatoire de Paris. Ces données, converties au format Python, permettent de déterminer la valeur de la masse du Soleil, puis, par un raisonnement analogue, celle de l'astéroïde.



D'après : https://media4.obspm.fr/public/ressources_lu/pages_lois-kepler/peser-kepler_impression.html

Retrouvez éduscol sur :



Récupération des données dans Python

Récupération du tableau de données

Sur la [page](#) de l'Observatoire de Paris « Peser l'Univers avec la troisième loi de Kepler », on trouve le tableau suivant :

La 3ème loi de Kepler appliquée au système solaire

Planète	a	T_{sid}	i	e	T^2/a^3
	UA	an	deg		an ² /UA ³
Mercure	0.3871	0.2408	7.0	0.206	0.9996
Vénus	0.7233	0.6152	3.4	0.007	1.0002
Terre	1.0000	1.0000	--	0.017	1
Mars	1.5237	1.8808	1.8	0.093	1.0000
Jupiter	5.2026	11.862	1.3	0.048	0.9992
Saturne	9.5547	29.457	2.5	0.056	0.9948
Uranus	19.218	84.020	0.8	0.046	0.9946
Neptune	30.109	164.77	1.8	0.009	0.9946

Les valeurs sont reportées dans un tableur intitulé [ActNum2donneesplanetes.xlsx](#). Le fichier doit être enregistré dans le même répertoire que le fichier python.

Conversion au format Python

La procédure est alors exactement la même que celle mise en œuvre dans la fiche précédente. Seules les données des colonnes 2 et 3 sont récupérées :

```
import openpyxl
import csv
FichierXLSX='donneesplanetes.xlsx'
wb = openpyxl.load_workbook(FichierXLSX)
ws = wb.worksheets[0]
NbLignes = ws.max_row
NumCol=2
a=[]
for x in range(NbLignes):
    valeur=ws.cell(row=x+1,column=NumCol).value
    a.append(valeur)
print(a)

NumCol=3
T=[]
for x in range(NbLignes):
    valeur=ws.cell(row=x+1,column=NumCol).value
    T.append(valeur)
print(T)
at=np.array(a)
tt=np.array(T)
```

Les deux dernières lignes ont pour but d'adapter le format du tableau récupéré pour en faire un objet manipulable par numpy.

Retrouvez éducol sur :



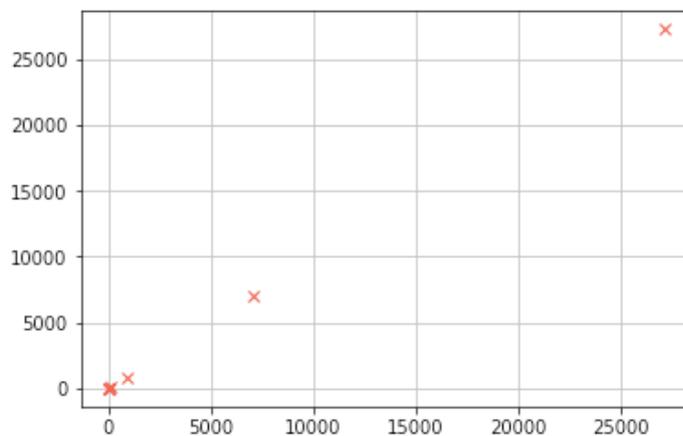
Exemples d'exploitation

Vérification d'une loi en physique

En traçant a^3 en fonction de T^2 , il est possible de tester la validité de la loi de Kepler. En seconde, on peut faire admettre aux élèves cette loi – en disant qu'elle est une simple conséquence du lien entre force et variation de vitesse et de l'expression vue du champ de gravitation. On peut utiliser cette loi pour en déduire une valeur de la masse du Soleil. Cela donne :

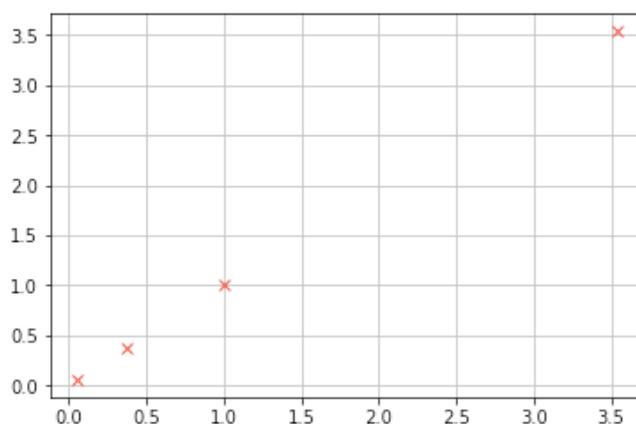
```
a3=(at)**3
t2=(tt)**2
plt.grid()
plt.plot(t2,a3,'rx')
```

qui renvoie :



Remarque : ce peut être l'occasion d'un travail sur les échelles de représentation. Ici, l'orbite conséquente de Pluton écrase toutes les autres données et ne permet pas d'apprécier la validité de la loi. En se limitant aux 4 premières planètes, le tracé devient :

```
att=at[:4]
t2=tt[:4]
a3=(att)**3
t2=(t2)**2
plt.grid()
plt.plot(t2,a3,'rx')
```



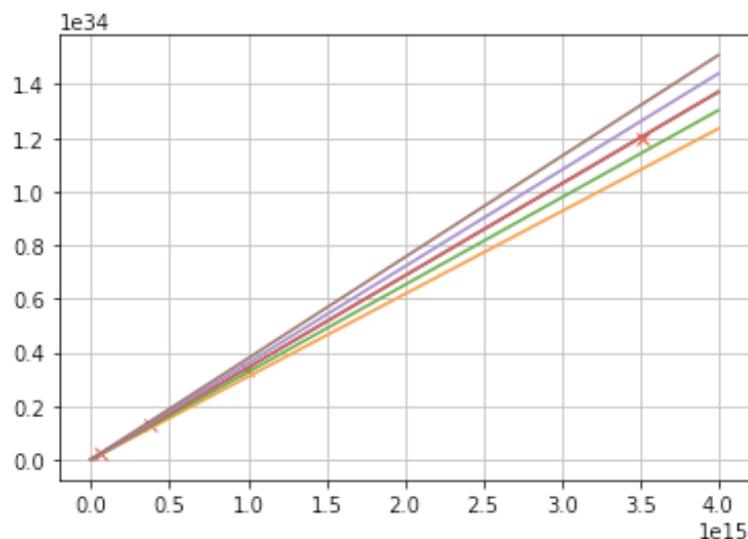
Retrouvez éducol sur :



La détermination du coefficient directeur de la courbe permet d'obtenir la masse du Soleil. Pour la détermination du coefficient directeur, plutôt que d'utiliser une procédure « boîte noire », il est possible de faire tracer aux élèves un ensemble de droites de pentes différentes, pour procéder par essai et erreur. C'est l'occasion de manipuler une boucle for. Comme les données ont été entrées en u.a. et en années, la pente est forcément égale à 1, mais l'on peut reprendre le fichier et tout convertir en unités du Système International, ce qui donne :

```
ar=att*(1.5*10**11)
tr=ttt*(365*24*3600)
a3=(ar)**3
t2=(tr)**2
plt.grid()
plt.plot(t2,a3,'rx')
l=linspace(0,4*10**15,20)
pente=arange(5)
y=(1.2*10**34/(3.5*10**15))*l
plt.plot(l,y)
def test(i):
    return ((1.2*10**34/(3.5*10**15))+0.05*(i-2)*(1.2*10**34/(3.5*10**15)))*l
for i in pente :
    plt.plot(l,test(i))
```

qui renvoie :



Modélisation automatique

La commande `np.polyfit` permet également, comme dans les fiches d'activités précédentes, de déterminer un modèle linéaire de la tendance de cet ensemble de points.

Utilisation d'une loi en physique

Munis de la loi de Kepler, les élèves peuvent ensuite reprendre l'étude de l'astéroïde Eugénie pour déterminer sa masse à partir du mouvement de son satellite.

Retrouvez éducol sur :

