


Terminale Spécialité Physique-Chimie	Thème : Mouvement et interactions	M.KUNST-MEDICA	
Chapitre 8 : Mouvement dans un champ de gravitation			

Feuille d'évaluation à rendre obligatoirement avec la copie

Activité numérique n°8.2 : Neptune et ses satellites

Inspiré de Belin éducation – Cahier Python-Arduino

	Questions	Compétence visée	Points attribués
Appel n°1	1	S'approprier	/1
	2		/0,5
Appel n°2	3	Réaliser	/0,5
	4		/0,5
Appel n°3	5	Réaliser, analyser, raisonner	/0,5
	6		/0,5
	7		/0,5
Appel n°4	8	Analyser, raisonner, communiquer	/0,5
Devoir global	Rendre compte à l'écrit en utilisant un vocabulaire scientifique adapté et présenter son travail sous une forme appropriée et être vigilant vis-à-vis de l'orthographe	Communiquer	/0,25
Total 1 :	Remarques :		/4,75

Notation individuelle :

CLASSE :		NOMS – PRENOMS des élèves du groupe		Élève n° 1 :		Élève n° 2 :		Élève n° 3 :	
				
				
Activité	Capacités attendues	Compétence visée	Points attribués	Signatures	Points attribués	Signatures	Points attribués	Signatures	
Séance en groupe	Travailler en équipe, partager des tâches, s'engager dans un dialogue constructif, respecter ses camarades, son professeur et les lieux de travail ...	Être autonome et faire preuve d'initiative	/0,25		/0,25		/0,25		
TOTAL 2			/0,25		/0,25		/0,25		
Total 1 + 2			/5		/5		/5		

Capacité numérique exigible : Exploiter, à l'aide d'un langage de programmation, des données astronomiques ou satellitaires pour tester les deuxième et troisième lois de Képler.

Annexes :

- **Savoir rentrer des valeurs numériques dans un programme Python- p10**
- **Utiliser un programme Python pour appliquer une formule- p10**
- **Utiliser un programme Python pour tracer un graphique- p30**
- **Reconnaître les instructions de base : plt, plot(), print(), round()- p12-14**

Neptune est la 8^e planète du système solaire par ordre d'éloignement du Soleil. Elle met 165 ans pour faire un tour autour du Soleil suivant une orbite quasi circulaire. Neptune possède 14 satellites naturels, parmi lesquels Naiade, Thalassa, Despina, Galatée, Larissa, Hippocampe et Protée suivent une orbite circulaire.

Johannes Kepler (1571-1630), astronome et mathématicien allemand, a énoncé trois lois relatives au mouvement des planètes autour du Soleil, qui peuvent être étendues aux satellites en orbite autour d'une planète.



Comment tester les lois de Képler dans le cas de Neptune ?

Partie 1 : Enoncé des lois de Kepler

1^{re} loi de Kepler ou loi des orbites :

Dans le référentiel héliocentrique, la trajectoire du centre P d'une planète est une ellipse dont l'un des foyers est le centre du Soleil.

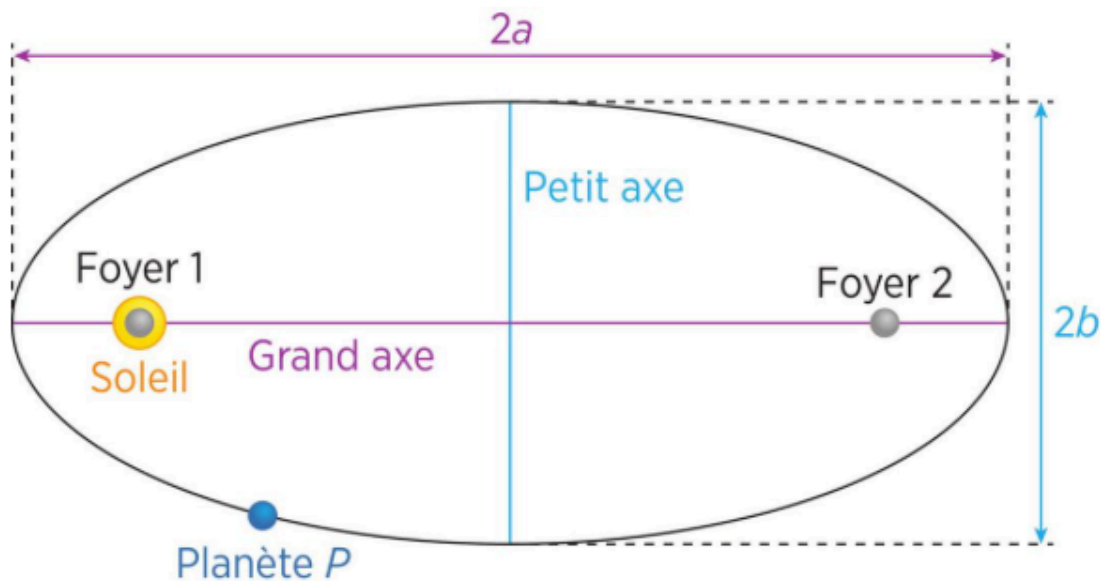


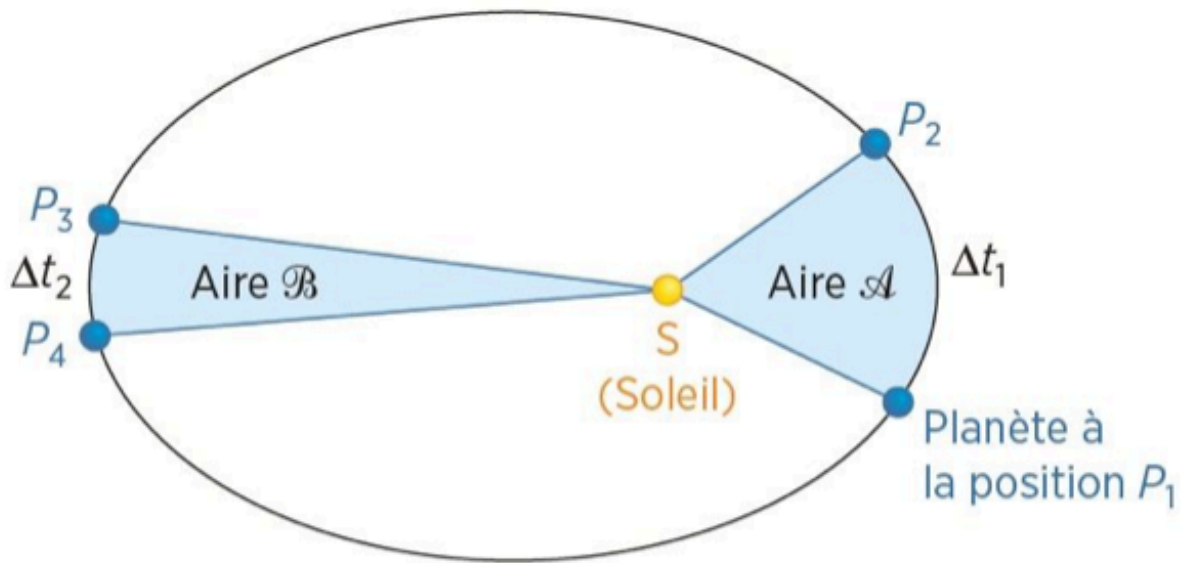
Schéma de la trajectoire elliptique d'une planète

C

© Boile Éducation/Humensis, 2021 Python et Arduino pour la physique-chimie - Spécialité
© STDI

2^e loi de Kepler ou loi des aires :

Le segment $[SP]$ qui relie le centre S du Soleil au centre P d'une planète balaie des aires égales pendant des durées Δt égales.



Si $\Delta t_1 = \Delta t_2$ alors

C

© Belin Éducation/Humensis, 2021 Python et Arduino pour la physique-chimie - Spécialité
© STDI

3^e loi de Kepler ou loi des périodes :

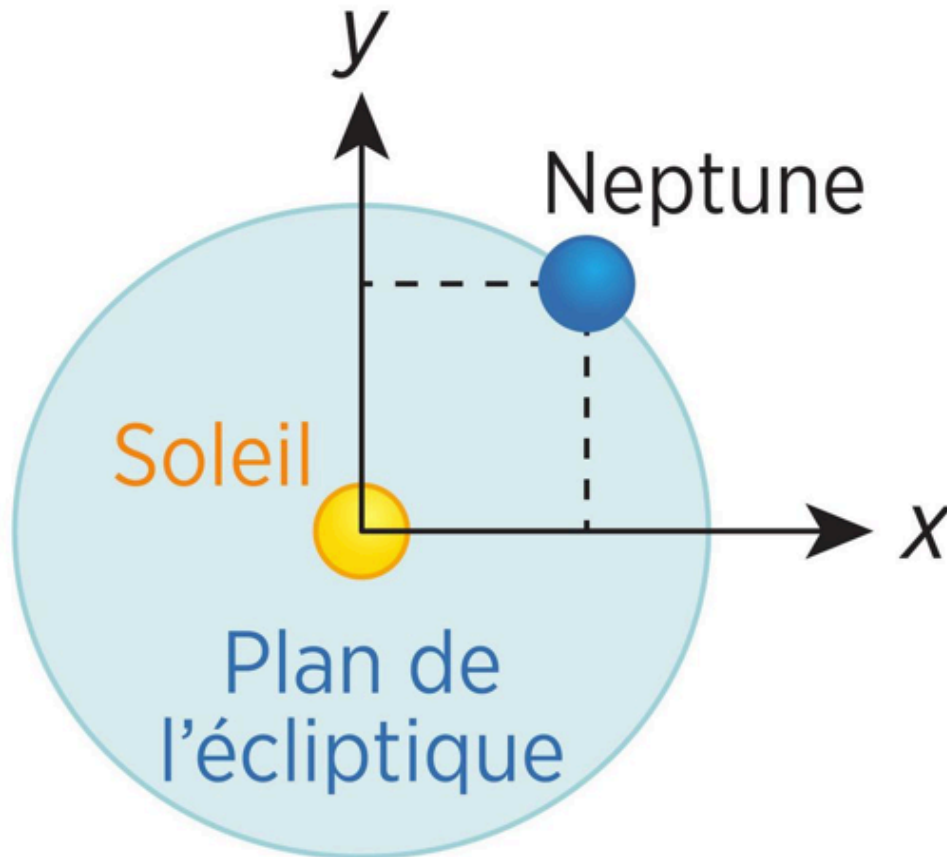
Le quotient du carré de la période de révolution T (en s) d'une planète par le cube de la longueur a (en m) du demi-grand axe de son orbite est égal à une même constante k (en) pour toutes les planètes du système

solaire : $\frac{T^2}{a^3} = k$.

Donnée : constante de gravitation universelle $G = 6,67 \times 10^{-11} \text{ N} \cdot \text{m}^2 \cdot \text{kg}^{-2}$.

Partie 2 : Ephéméride des positions de Neptune dans le repère héliocentrique éclipstique

La position prévisionnelle de Neptune dans le repère héliocentrique éclipstique est donnée dans le tableau ci-dessous (à télécharger) à la même date entre 2020 et 2024 (avec pour référence l'éclipstique du 1^{er} janvier 2000).



© Belin Éducation/Humensis, 2021 Python et Arduino pour la physique-chimie - Spécialité
© STDI

Donnée : 1 ua = $1,5 \times 10^{11}$ m = distance

Année	2020	2021	2022	2023	2024
Abscisse x (ua)	29,4328410560730	29,6118356847488	29,7457682031831	29,8342404541355	29,8774316380940
Ordonnée y (ua)	- 5,3906071859853	- 4,2505317595919	- 3,1034908723970	- 1,9519995345194	- 0,7985426911486

Propriétés mathématiques :

• Formule de Héron d'Alexandrie :

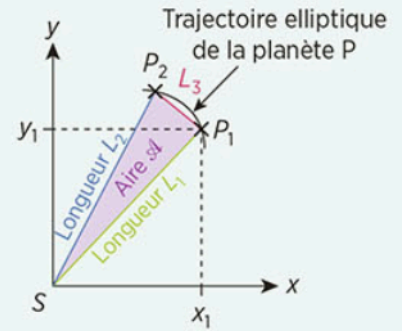
Calcul d'une aire balayée par le segment [SP] entre deux points P_1 et P_2 :

$$\text{Aire } \mathcal{A} = \sqrt{p \times (p - L_2) \times (p - L_3) \times (p - L_1)}, \text{ avec } p = \frac{L_1 + L_2 + L_3}{2}.$$

• Calcul de l'excentricité e d'une ellipse :

$$e = \sqrt{1 - \frac{b^2}{a^2}} \text{ avec } a \text{ la longueur du demi-} \dots \text{ axe de l'ellipse et } b \text{ la longueur de son demi-} \dots \text{ axe.}$$

L'excentricité indique l'écart au cercle de la forme de l'orbite : une excentricité nulle correspond à un cercle.



Application

Soient P_1 et P_2 les positions prises par Neptune respectivement en 2020 et en 2021 (tableau 2).

• Compléter les coordonnées :

– du point P_1 (..... ;) ;

– du point P_2 (..... ;) ;

– du Soleil S (..... ;).

• Calculer la longueur L_1 entre le Soleil et P_1 :

.....

• Calculer la longueur L_2 entre le Soleil et P_2 :

.....

Partie 3 : Caractéristiques des satellites de Neptune

Le tableau ci-dessous donne les valeurs du demi-grand axe a de l'orbite des satellites réguliers de Neptune (c'est-à-dire ayant une orbite quasi circulaire) et leur période de révolution T .

Satellite	Naiade	Thalassa	Despina	Galatée	Larissa	Hippocampe	Protée
a (km)	$48,227 \times 10^3$	$50,075 \times 10^3$	$52,526 \times 10^3$	$61,593 \times 10^3$	$73,458 \times 10^3$	$105,200 \times 10^3$	$117,647 \times 10^3$
T (j)	0,294	0,311	0,335	0,429	0,555	0,936	1,122

Questions :

S'approprier , Réaliser, analyser - raisonner

1. **Compléter** les parties 1 et 2.

2. L'excentricité de Neptune $e = 0,00896$ est-elle en accord avec sa trajectoire quasi circulaire autour du Soleil ?

.....

Appel n°1 du professeur pour validation

Partie 4 : Vérification de la 2^{ème} loi de Kepler

1 à 3

6 à 8 Création de trois listes contenant les valeurs des X , des Y et les t correspondant aux positions de Neptune

11 à 14 Création et paramétrage du graphique $Y = f(X)$, qui correspond à une

15

18 à 23 Calcul et affichage des aires balayées entre 2020 et 2021

26 à 31

```

1 import matplotlib.pyplot as plt # importe les bibliothèques
  nécessaires
2 import numpy as np
3 import math as m
4
5 # Coordonnées de Neptune en unité astronomique
6 X = [ _____ , _____ , _____ , _____ ] # abscisse
7 Y = [ _____ , _____ , _____ , _____ ] # ordonnée
8 t = [ _____ , _____ , _____ , _____ ] # dates
  de relevé des coordonnées
9
10 # Représentation d'une portion de l'orbite de Neptune de 2020 à 2024
  dans le plan de l'écliptique
11 plt.plot(X, Y) # trace la courbe X en fonction de Y
12 plt.xlabel('_____') # légende de l'axe
  des abscisses
13 plt.ylabel('_____')
14 plt.title("Portion d'orbite de Neptune dans le référentiel
  héliocentrique") # titre du graphique
15 plt.show() # affiche le graphique
16
17 # Calcul des aires parcourues entre 2020 et 2021
18 L1_a = m.sqrt(X[0]**2 + Y[0]**2) # distance L1_a en ua
19 L2_a = m.sqrt(X[1]**2 + Y[1]**2)
20 L3_a = m.sqrt((X[1] - X[0])**2 + (Y[1] - Y[0])**2)
21 p_a = (L1_a + L2_a + L3_a)/2 # demi-périmètre
22 Aire_a = m.sqrt(p_a*(p_a - L2_a)*(p_a - L3_a)*(p_a - L1_a))
  # aire balayée entre 2020 et 2021
23 print ("L'aire parcourue par le rayon vecteur entre 2020 et
  2021 vaut : ", round(Aire_a, 2), "ua^2") # affiche le résultat
24
25 # Calcul des aires parcourues entre 2022 et 2023
26 L1_b = m.sqrt(X[_____]**2 + Y[_____]**2) # distance L1_b en ua
27 L2_b = m.sqrt(X[_____]**2 + Y[_____]**2)
28 L3_b = m.sqrt(_____ )
29 p_b = _____ # demi-périmètre
30 Aire_b = m.sqrt(p_b*(_____ - L2_b)*(p_b - L3_b)*(p_b
  - _____)) # aire balayée entre 2022 et 2023
31 print ("L'aire parcourue par le rayon vecteur entre 2022 et
  2023 vaut : ", round(_____, 2), "ua^2") # affiche le
  résultat
          
```

• La fonction plt.plot(_____, _____) affiche la courbe $Y = f(X)$ avec des points expérimentaux reliés.

• La fonction print(grandeur) affiche _____

• La fonction sqrt(grandeur) calcule _____

Pour une liste notée X, l'instruction X[0] appelle _____

La fonction round(valeur, 2) permet d'arrondir la valeur au _____

⇒ Voir cours Partie 1

COUP DE POUCE

La norme d'un vecteur $\vec{u}(x; y)$ est donnée par la formule :

$$\|\vec{u}\| = \sqrt{x^2 + y^2}$$

Réaliser

- On souhaite tracer une portion de trajectoire de Neptune et vérifier que la 2^{ème} loi de Kepler s'applique bien à cette planète. Dans le programme Python ci-dessus, **compléter** :
 - Les valeurs contenues dans les listes X, Y et t (lignes 6, 7 et 8) ;
 - Les légendes des axes du graphique (lignes 12 et 13) ;
 - Les calculs de L1_b, L2_b, L3_b, p_b et Aire_b (lignes 26 à 30) ;
 - L'affichage de l'aire (ligne 31)

4. **Exécuter** le programme mis à disposition. La deuxième loi de Kepler est-elle vérifiée ? justifier.

Appel n°2 du professeur pour validation

Partie 5 : Vérification de la 3^{ème} loi de Kepler

2 à 4 Importation des bibliothèques et des modules

7 et 8 Création de deux tableaux contenant les _____

9 et 10 Calcul de _____

13 à 16 Création et paramétrage du graphique _____ = f(_____)

19 et 20 Régression linéaire et affichage

21 Calcul de _____ modélisé connaissant R³ et a

22 Création de la _____

23 et 24 Affichage du graphique

Pour vérifier la 3^e loi de Kepler dans le cas des satellites de Neptune, exécuter :

```
1 # Importation des bibliothèques et modules
2 import matplotlib.pyplot as plt
3 import numpy as np
4 from scipy.stats import linregress # importe le module linregress
5
6 # Création des tableaux de valeurs
7 R = np.array([48.227e3, _____, _____,
8 _____, _____, _____])
9 # valeurs des rayons (en km) des orbites circulaires des satellites autour de Neptune
10 T = np.array([_____, _____, _____, _____, _____, _____]) # période de révolution (en jour) des satellites de Neptune
11 R3 = R**3 # cube du rayon de l'orbite du satellite
12 T2 = _____ # carré de la période
13 # Tracé de la courbe expérimentale T2 en fonction de R3
14 plt.plot(R3, T2, '+') # trace les points expérimentaux pour la courbe T^2 = f(R^3)
15 plt.xlabel("_____") # légende de l'axe des abscisses
16 plt.ylabel("_____")
17 plt.grid() # affiche un quadrillage
18 # Régression linéaire et affichage de la droite de régression
19 (a, __, __, __, __) = linregress(R3, T2) # modélise la courbe précédente par une droite de coefficient directeur a
20 print('Le coefficient directeur de la droite est : ', a, "en jour^2/km^3") # affiche la valeur du coefficient directeur a
21 T2_modelisation = a*R3 # calcule la période connaissant le coefficient directeur a
22 plt.plot(R3, T2_modelisation, label = 'Droite de régression linéaire') # trace la droite de régression linéaire
23 plt.legend() # affiche la légende des courbes
24 plt.show() # affiche le graphique
```

La fonction plt.plot(abscisse, ordonnée, '+') permet _____

(a, b, __, __, __) = _____(x, y) permet de réaliser une régression linéaire pour les points expérimentaux de la courbe y = f(x). a correspond au _____ et b à _____

➔ Voir cours Partie 1

Réaliser

5. On souhaite vérifier que la 3^{ème} loi de Kepler s'applique bien aux satellites naturels de Neptune. A l'aide des données du tableau partie 3, **compléter** dans le programme Python ci-dessus les deux tableaux de valeurs correspondant à la période T de révolution des satellites et au rayon R de leur orbite considérée comme circulaire.

6. **Compléter** le calcul T^2 (ligne 10) et les légendes des axes du graphique $T^2 = f(R^3)$ (lignes 14 et 15).
7. **Exécuter** le programme mis à disposition. La 3^{ème} loi de Kepler est-elle vérifiée ? Justifier

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Appel n°3 du professeur pour validation

Analyser

8. A partir des résultats du programme Python précédent, en déduire la masse de Neptune, sachant que :

$$\frac{T^2}{R^3} = \frac{4 \times \pi^2}{G \times M_{\text{Neptune}}}$$

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Appel n°4 du professeur pour validation