**Les lois de Kepler**



Johannes Kepler (1571-1630)

*Johannes Kepler a découvert de manière empirique (en s’appuyant sur les observations très fines d’un autre scientifique : Tycho Brahé) les lois qui régissent le mouvement des planètes. Il lui a fallu 15 ans pour les énoncer. Ces lois ont été démontrées plus tard par Newton.*

*Copier les 4 fichiers suivants dans votre dossier personnel :*

***Kepler\_1ere\_loi.py******Kepler\_2eme\_loi.py******Terre2019.csv******Kepler\_3eme\_loi.py***

*Le code « Kepler\_2eme\_loi » est proposé par Guillaume Dewaele, professeur en CPGE à Anger – Crédit Photo : Wikipedia – schémas : O. Chaumette*

**1. Première loi de Képler**

***1.1. Récupération de données astronomiques :***

Suivre la notice présente sur la table et sauvegarder le fichier csv dans votre dossier personnel sous le nom « **Mercure2019.csv** » (avec un M majuscule et ne pas oublier «  .csv »)

***1.2. Travail sur feuille :***

Le site de données astronomiques donne, pour chaque date, les positions de Mercure dans le référentiel héliocentrique dans un système de coordonnées dit « polaire » :

- **r** est la distance Soleil-Mercure.

- **θdeg** est l’angle en degrés entre l’axe Sx du repère du référentiel et le vecteur Soleil-Mercure.

**1.** Exprimer θ (en radian) en fonction de θdeg (valeur de θ en degrés)

**2.** Exprimer les coordonnées cartésiennes x et y de Mercure en fonction de r et θ.

***1.3. Modification d’un programme Python permettant de tracer la trajectoire de Mercure :***

**1.** Ouvrir le logiciel Edupython et charger le programme **Kepler\_1ere\_loi.py** **de votre dossier personnel**.

**2.** En dessous de la partie rose « *Début du Travail des élèves* », compléter les formules permettant de calculer **teta** (θ) à partir de **teta\_deg** (θdeg) ainsi que **x** et **y**.

En python, π se tape : **np.pi** et cos(x) se tape : **np.cos(x)**

**3.** Taper le code permettant de tracer y en fonction de x, avec des croix rouges par exemple. Exécuter le programme.



***1.4. Observation et énoncé de la 1ère loi de Kepler :***

**1.** Que peut-on dire de la trajectoire de Mercure ? Où se situe le Soleil ? L’excentricité de la trajectoire est-elle proche de 1 ou proche de 0 ?

**2.** Les observations faites pour Mercure peuvent se généraliser à toutes les planètes. Enoncer la 1ère loi de Kepler.

**3.** Remplacer les lignes  : **planete="Mercure"** et **Nom\_fichier="Mercure2019.csv"**

par : **planete="Terre"** et **Nom\_fichier="Terre2019.csv"** et exécuter le programme.

**4.** Que dire de la trajectoire de la Terre ? L’excentricité de la trajectoire est-elle plus faible que pour Mercure ou plus grande ? Rechercher sur internet l’excentricité de l’orbite de la Terre.

**2. Seconde loi de Képler :**

**1.** Ouvrir dans Edupython le programme **Kepler\_2eme\_loi.py** **de votre dossier personnel**. L’exécuter.

L’animation dessine les aires A1 et A2 parcourues par le rayon soleil-planète pendant une même durée. On peut mettre en pause en cliquant sur la figure.

**1.** Que penser des aires A1 et A2?

**2.** Enoncer la 2nde loi de Kepler.

**3.** A quel moment la planète va-t-elle le plus vite autour du Soleil ?

**4.** Modifier dans le code python l’excentricité de l’orbite pour que cette dernière soit celle de la Terre. La vitesse varie-t-elle le long de l’orbite ?

**5.** Créer enfin une orbite circulaire. Que ce se passe-t-il pour la vitesse?

**6.** ***Culture générale :*** quand la planète est au plus proche du soleil, on dit qu’elle est au **périhélie**. Quand elle est le plus éloignée sur l’orbite, elle est à **l’aphélie**.

Peut-on simplement prévoir à quelle saison la Terre se trouve au périhélie ?

**3. Test de la troisième loi de Kepler**

Kepler a observé que **la période de révolution *T*** d’une planète était d’autant plus grande que **sa distance moyenne *R* au Soleil** était grande. Plus tard, Newton a montré que  donc que  est constant quelle que soit la planète.

T : période de révolution de la planète (en s), R : distance moyenne au soleil (en m) et MS :masse du Soleil.

***Données***: G=6,6710-11 m3.kg−1.s−2 et MSoleil = 21030 kg

**1.** Ouvrir dans Edupython le programme **Kepler\_3eme\_loi.py** **de votre dossier personnel**.

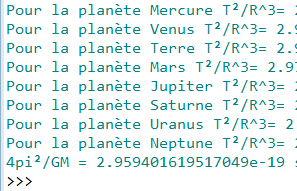
*Le programme Python proposé comporte 3 tableaux : le nom des planètes (****planete****), leur période de révolution (****T****) en année et leur distance moyenne au soleil (****R****) en millions de km.*

**2. REA** Suivre les consignes situées dans le code au niveau de « **TRAVAIL 1** »convertir T et R dans les unités du système international.

**AIDE** En python, pour convertir tout un tableau, il suffit de le multiplier par la constante adéquate.

*Exemple :* **Lambda=[400,500,600,700] # longueur d’onde en nanomètres**

**Lambda = Lambda\*1e-9 # conversion du tableau Lambda en mètres**



***Extrait du résultat attendu***

**Niveau EXPERT**

**3. REA** Réaliser une partie de programme Python, en dessous de « **TRAVAIL 2** »**,** permettant de montrer que T²/R3 est égal à 4π²/(G.MS) pour chaque planète. Il est, bien sûr, hors de question de faire 8 calculs de suite !

**4. COM** Mettre en forme au mieux les résultats de vos calculs.

**Niveau AVANCÉ**

**3. REA** En dessous de « **TRAVAIL 2** » **:**

**a.** Mettre la valeur de 4π²/(G.MS) dans une variable

**b.** Créer un nouveau tableau contenant les valeurs de T²/R3 de chaque planète.

**4. COM** Mettre en forme au mieux les résultats de vos calculs : on pourra utiliser l’algorithme suivant :

- mettre dans une variable NP le nombre de planètes du tableau « planète »

- faire une boucle : ***n*** variant de 0 à NP :

Afficher : "***Pour la planète***" suivi du nom de la planète en position ***n*** du tableau « planete », "***T²/R^3 vaut***" suivi de la valeur de T²/R3 de la planète

- Afficher enfin la valeur de 4π²/(G.MS)

**Niveau DÉBUTANT**

**3. REA** En dessous de « **TRAVAIL 2** » **:**

**a.** Assigner aux variables G et Ms les bonnes valeurs et mettre la valeur de 4π²/(G.MS) dans une variable appelée K

**b.** Créer un nouveau tableau nommé « **T2\_sur\_R3**  » égal à T²/R3.

**4. COM** Mettre en forme au mieux les résultats de vos calculs : traduire en python l’algorithme suivant :

- faire une boucle : ***n*** variant de 0 à 8:

Afficher : "***Pour la planète***" suivi du nom de la planète en position ***n*** du tableau « **planete** », "***T²/R^3 vaut***" suivi de la valeur de T²/R3 située en position ***n*** du tableau « **T2\_sur\_R3**  ». **AIDE** Le nom de la planète en position ***n*** est : **planete[n]**

- Afficher enfin la valeur de 4π²/(G.MS) **AIDE** C’est à dire : afficher la valeur de K