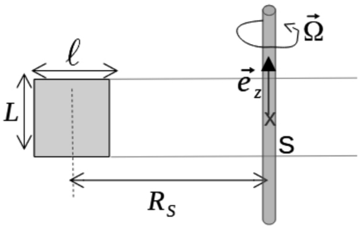
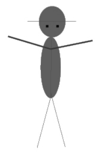


Figure 11 bis : détail de la vue en coupe



Figurine à placer

Q20. Évaluer l'ordre de grandeur de Ω afin que la valeur de g_a soit proche de celle de la valeur terrestre.

Q21. Donner une raison pour laquelle la pesanteur artificielle n'est pas utilisée dans l'actuelle station spatiale internationale ISS.

Partie III - Peser la Terre

Cette partie fait appel à de la programmation en langage Python, dont un certain nombre de " fonctions " sont rappelées en fin d'énoncé pages 17 à 19.

III. 1 - Principe

On dit que l'interaction gravitationnelle permet de peser les astres.

Q22. Justifier que l'action " peser un astre " est l'action de mesure de sa masse et non de son poids.

L'observation de satellites terrestres artificiels permet de déterminer la masse terrestre M_T.

On rappelle que ces satellites ont des masses très inférieures à celle de la Terre et des trajectoires *a priori* elliptiques dont un foyer est le centre d'inertie de la Terre O_T. On utilise pour cela la relation, établie par Kepler en 1619 dans l'ouvrage *Harmonices Mundi*, connue sous le nom de troisième loi de Kepler : $\frac{a^3}{T^2} = \frac{G\,M_T}{4\pi^2}$. Dans cette expression, a est le demi-grand axe de l'ellipse trajectoire et T la période de révolution du satellite.

Q23. Faire un <u>schéma clair</u> de l'ellipse, trajectoire du satellite, en y positionnant le foyer O_T et le demi-grand axe a. Faire figurer également les points particuliers P et A de la trajectoire, appelés respectivement «périgée» et «apogée», ainsi que les distances r_P et r_A de ces points au foyer O_T. Exprimer le demi-grand axe a en fonction de r_P et r_A.

On veut obtenir la valeur de M_T grâce à une régression linéaire à partir des couples de valeurs (a, T) obtenues pour 9 satellites différents. Pour cela, on trace la courbe $Y = F(X)$. On s'attend à une droite dont la pente, notée alpha, permet de calculer M_T.

Q24. Exprimer Y en fonction de a et X en fonction de T. Exprimer littéralement M_T en fonction de alpha et des valeurs constantes apparaissant dans la troisième loi de Kepler.

Par la suite, on attend un résultat en kg pour la valeur numérique de M_T.

III. 2 - Étude de données orbitales

On trouve sur le web plusieurs sites (par exemple le site *Live realtime satellite tracking and predictions*) qui donnent en temps réel les positions de centaines de satellites dans le référentiel terrestre.

On suit 9 satellites. Pour un satellite, on dispose alors de données telles que sa période (en minutes), et, en temps réel pour N_d dates, son altitude (en km), ainsi que sa latitude et sa longitude, en coordonnées terrestres. Après extraction des données, on obtient, pour le groupe des 9 satellites, une liste DATA composées de 9 listes. Chacune des 9 listes contient, pour chaque satellite, les valeurs suivantes : numéro, nom, T, t, lat, long, alt.

Description du groupe de listes pour un satellite donné :

Liste :	*description :*	*format :*
numero	numéro du satellite étudié	entier de 1 à 9
nom	nom du satellite	chaîne de caractères
T	période, en minute	flottant positif
t	liste des dates de mesures des positions,	liste de N_d flottants positifs
lat	liste des latitudes, en degré,	liste de N_d flottants
long	liste des longitudes, en degré,	liste de N_d flottants
alt	liste des altitudes, en kilomètre,	liste de N_d flottants positifs

On considère les instructions suivantes (document 1) :

```
n_sat=[ ]
for i in range(9) :
        n_sat.append(DATA[i][0])
```

Document 1

Q25. Expliquer ce que fait ce morceau de code.

Q26. Écrire les instructions pour extraire de DATA la liste T_sat contenant les 9 périodes exprimées en minutes.

Q27. Écrire la fonction **demiGrandAxe**(DATA) qui permet d'obtenir la liste a_sat contenant les demi-grands axes a des 9 trajectoires exprimées en km.

On obtient alors les listes décrites dans le **document 2** :

```
n_sat =  [ 1      , 2      , 3      , 4      , 5      , 6      , 7     , 8      , 9 ]
         # Numéro du satellite

T_sat =  [ 92.9  ,1436.1  ,101.9  ,1436.2  ,631.1  ,1214.4  ,93.   ,118.4  ,104.2 ]
         # Période de révolution T en minutes

a_sat =  [ 794.  ,42164.  ,7228.  ,42167.  ,24371.  ,37704.  ,6825.  ,7988.  ,7333.]
         # Demi-grand axe a en km
```

Document 2

Q28. Écrire la fonction **XY** dont les arguments sont les listes T_sat et a_sat, qui retourne les tableaux de type numpy.ndarray, X et Y dans le tuple (X,Y).

On calcule les paramètres alpha et beta de la droite de régression linéaire obtenues à partir des valeurs X et Y, afin d'obtenir le tableau Y_mod tel que : Y_mod = alpha * X + beta.

Q29. Écrire les instructions pour obtenir le tableau [alpha, beta] des paramètres de la régression linéaire. Quelle valeur attend-on pour l'ordonnée à l'origine beta ? Justifier.

Grâce à la bibliothèque **matplotlib.pyplot**, le code du document 3 permet de tracer Y = F(X) pour les 9 satellites enregistrés ainsi qu'une droite de régression linéaire.

```
plt.figure()
plt.plot( A , B ," bo ", label= " Données ")
plt.plot( C , D ," r- ", label= " Régression linéaire ")
plt.title(" Exploitation de la troisième loi de Képler ")
plt.xlabel( " E " )
plt.ylabel( " F " )
plt.legend()
plt.grid(True)
plt.show()
```

Document 3

Q30. Expliciter les termes **A**, **B**, **C**, **D**, **E**, et **F**.

Un élève obtient la courbe de la figure 12 et s'en satisfait.

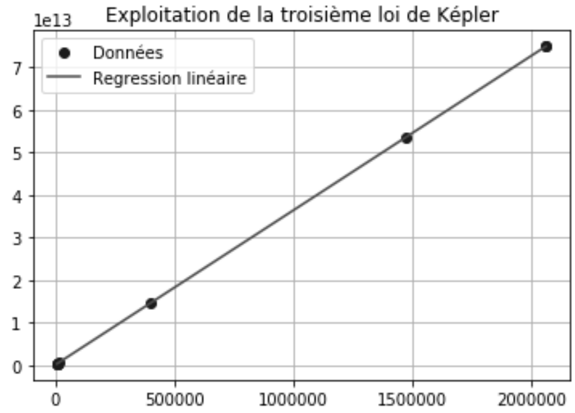


Figure 12 : graphique d'un élève

Hélas, ce graphique n'est pas validé par le professeur, pour plusieurs raisons.

Q31. Quelle(s) précaution(s) l'élève a-t-il oubliée(s) de prendre en écrivant la fonction **XY**, puis en écrivant le code décrit sur le document 3 ?

Q32. Expliquer pourquoi seuls 4 points semblent apparaître sur le graphique, alors que les 9 valeurs ont bien été entrées dans le code.

III. 3 - Précision du résultat

Cette méthode permet une première valeur pour M_T.

Les données précédemment utilisées sont connues avec une " marge " e_val, c'est-à-dire qu'une valeur associée à la mesure val se situe dans l'intervalle $[\text{val} - \text{e\_val}\ ,\ \text{val} + \text{e\_val}]$. Il est donc légitime de s'intéresser à la précision du résultat obtenu pour M_T.

Afin de comparer la compatibilité de la valeur numérique obtenue avec la masse de la Terre publiée, on évalue l'incertitude-type u_M grâce à une simulation de Monte-Carlo. Pour cela, pour les 9 satellites retenus, on simule N_sim mesures de a et T. Les résultats de ces mesures sont répartis uniformément dans les intervalles $[\text{a} - \text{e\_a}\ ,\ \text{a} + \text{e\_a}]$ et $[\text{T} - \text{e\_T}\ ,\ \text{T} + \text{e\_T}]$. On estime $\text{e\_a} = 0.01\ \% \times \text{a}$ et $\text{e\_T} = 0.01\ \% \times \text{T}$.

Cette simulation procède à N_sim tirs aléatoires générés par une fonction Python de la bibliothèque **numpy.random**. Chaque tir aléatoire donne deux valeurs T_tir et a_tir, comprises dans les intervalles définis par la précision des mesures.

Pour chaque couple de valeurs tirées, on calcule les valeurs Y_tir et X_tir associées, puis les paramètres de la régression linéaire : la valeur de la pente, alpha_tir, et l'ordonnée à l'origine, beta_tir.

Finalement, les N_sim valeurs obtenues pour alpha_tir et beta_tir, permettent d'accéder, grâce à leurs moyennes et leurs écart-types, à M_T, u_M, beta_sim, et l'incertitude-type associée à beta, u_beta.

Q33. Proposer un ordre de grandeur pour N_sim.

La séquence d'instructions permettant d'obtenir les listes alpha_tir, beta_tir, puis M_tir, peut s'écrire comme décrit ci-dessous (document 4) :

```
# Initialisation ds listes
list_MT, list_beta, list_alpha = G, H, I
list_Y, list_X = J, K

# Tirages aléatoires
for i in range(L) :
        a_tir= M + (e_a) * rd.uniform(-1, 1, 9)        # Valeur aléatoire de a
        T_tir= N + (e_T) * rd.uniform(-1, 1, 9)        # Valeur aléatoire de T
        Y_tir= O
        X_tir= P

        list_Y.append(Q)          # AA
        list_X.append(R)

        p=np.polyfit(S,U,1)          # Régression linéaire
        list_alpha.append(V)
        list_beta.append(W)
        list_MT.append(Z)
```

Document 4

Q34. Expliciter les termes **G**, **H**, **I**, **J**, **K**, **L**, **M**, **N**, **O**, **P**, **Q**, **R**, **S**, **U**, **V**, **W**, **X**, **Y**, et **Z**. Proposer également le commentaire **AA**.

Q35. Écrire alors les instructions pour obtenir la masse M_T, l'incertitude-type associée u_M, ainsi que beta_sim et l'incertitude-type associée u_beta.

Le document 5 reproduit les résultats tels qu'ils sont affichés par le programme.

M_T = 5.9758189595337378e + 24 kg
u_M = 7.708549319930416e + 20
beta_sim = −3.388893817989048e + 17
u_beta = 7.923053822786523e + 17

Document 5

On compare le résultat obtenu, M_T, avec la valeur publiée M_T. La littérature mentionne : M_T = 5.97223 e+24 kg, u_T = 5.9 e+20 kg.

Q36. Définir un critère de compatibilité. Les résultats obtenus (valeur et précision) à partir des données de DATA sont-ils compatibles avec la valeur publiée ?