

Semaine 7 : du 13 au 17 octobre

La colle commence par une application de cours extraite de la liste ci-dessous et se poursuit par un exercice.

Je rappelle que vous trouverez sur mon site la version complétée du poly de cours, ainsi que les corrigés des TD et des DM. N'hésitez surtout pas à me signaler tout lien manquant ou défectueux.

Au programme

Chapitre M1 : Lois de Newton

Applications de cours et exercices.

Chapitre O2 : Instruments d'optique

Applications de cours et exercices.

Outils pour la physique : Incertitudes

Aucun développement théorique, ni aucun exercice spécifique, en revanche une question d'incertitudes est la bienvenue au sein d'un exercice.

Tout le contenu de la fiche outil (sauf la formule de composition d'incertitudes dans le cas d'un produit) a été abordé au moins une fois ... mais une fois seulement : on gardera donc des ambitions raisonnables, en particulier sur les simulations Monte-Carlo.

Outils pour la physique : Utilisation de Python

Désormais, les étudiants doivent être en mesure de compléter ou d'écrire entièrement un code Python simple (méthode d'Euler, traitement de données expérimentales, tracé de courbes). On utilisera (exclusivement) les bibliothèques numpy et matplotlib.

Applications de cours

Ces applications de cours sont des « briques élémentaires » des raisonnements à mener dans les exercices : les maîtriser est incontournable. Elles sont toutes traitées de manière exhaustive dans le cours.

Le travail demandé consiste à se les approprier, afin d'être capable de les réinvestir dans un sujet d'écrit ou d'oral. Je n'attends pas des étudiants un apprentissage par cœur, mais j'attends qu'ils les aient travaillées suffisamment pour les mener à bien en autonomie, c'est-à-dire savoir refaire seul les raisonnements, sans aide de l'interrogateur.

M1.1 - Établir l'équation du mouvement d'une chute libre, et la résoudre pour établir les lois horaires $(x(t), z(t))$.
Les intégrations seront menées par séparation de variables.

M1.2 - Les équations horaires d'un mouvement de chute libre sont données par

$$\begin{cases} x(t) = v_0 \cos \alpha t \\ z(t) = -\frac{1}{2}gt^2 + v_0 \sin \alpha t \end{cases}$$

En déduire l'équation de la trajectoire, la portée et la flèche du tir.

M1.3 - Établir l'équation différentielle vérifiée par la vitesse v d'un parachutiste chutant à la verticale dans l'air, en considérant une force de frottement quadratique. Adimensionner l'équation différentielle pour identifier la vitesse limite v_∞ et le temps caractéristique τ nécessaire pour atteindre cette vitesse.

M1.4 - Écrire un code Python permettant de résoudre par la méthode d'Euler explicite l'équation différentielle adimensionnée

$$\frac{dv}{dt} + v^2 = 1 \quad \text{avec} \quad v(0) = 0.$$

Écrire le code permettant de représenter la solution obtenue.

- Avant toute écriture de code, on commencera par établir la relation de récurrence pertinente.
- L'étudiant devra définir les paramètres de la simulation (pas de temps et durée totale adimensionnés), et construire correctement les listes t et v .

O2.1 - Tracer la marche des rayons issus d'un objet à l'infini dans une lunette astronomique. Définir le grossissement et établir (= démontrer) son expression en fonction des focales des lentilles.

O2.2 - Rappeler la modélisation d'un appareil photo. Justifier l'influence de la focale de l'objectif sur le champ.

O2.3 - Rappeler la modélisation d'un appareil photo. En raisonnant sur des schémas qualitatifs¹ (mais raisonnablement propres quand même!), justifier l'influence de l'ouverture du diaphragme sur la profondeur de champ.

À quoi s'attendre pour les programmes suivants ?

- Chapitre C1 : Cinétique chimique ;
- Chapitre E3+M2 : Régime libre des oscillateurs.

1. Les schémas que j'attends sont du type de la figure 16 du cours. Construire géométriquement la profondeur de champ (application 3) à main levée au tableau me semble trop difficile, mais vous devez savoir le faire sur une feuille quadrillée si un exercice le demande.