

## Présentation EDUSCOL

### Cadre de cette activité :

- Contexte : Simuler l'interaction d'objets (chocs élastiques) en deux dimensions avec éventuelle présence de forces à distances type gravité et tracer leurs trajectoires au cours du temps.
- Objectifs : Mettre en œuvre la méthode d'Euler pour la résolution du déplacement des balles sous l'effet des forces en présence (autres balles et contour) à l'aide du principe fondamental de la dynamique. L'extension en 3D est relativement simple ensuite mais non proposée.
- Matière : Sciences de l'Ingénieur
- Niveau : CPGE SPE : MP-PC-PT-PSI
- Activité : TD réalisé en classe généralement en 2h.
- Séquence : Chapitre « Ingénierie numérique » - Euler
- Matériel : Ordinateur avec logiciel de programmation en langage Python (ex : Pyzo+Anaconda). Modules random, math et matplotlib. Eventuellement ffmpeg et os pour la génération de vidéos depuis Python.
- Compétence du programme de SI :
  - o C3.1.3 - Choisir une démarche de résolution d'un problème d'ingénierie numérique ou d'intelligence artificiel. S3
  - o C3.3.1 - Mener une simulation numérique. S4
  - o C3.3.2 - Résoudre numériquement une équation ou un système d'équations. S3
- Ressources mises à disposition :
  - o Cours + Résumé « Euler »
  - o Sujet
  - o Code élèves et code corrigé
  - o Vidéos de simulations

### Remarques :

- J'ai laissé dans le sujet les liens vers des vidéos que j'ai publiées sur Dailymotion, mais les vidéos sont aussi à disposition en direct dans les documents partagés
- Un code élèves est proposé pour une prise en main rapide, avec :
  - o La proposition d'utiliser la programmation objet (sans avoir à s'y connaître) pour manipuler très simplement les données des balles par des appels du type `balle.ax` pour son accélération suivant x, `balle.x` pour son abscisse, `balle.r` pour son rayon etc.
  - o Une fonction permettant de gérer l'affichage du système
- Les animations obtenues sont plaisantes, les élèves adorent, moi aussi !
- On peut ajouter la gravité, modifier les paramètres des balles, ajouter des forces visqueuses etc. On pourrait ensuite aller plus loin en ajoutant le calcul des énergies cinétiques etc.
- Outre la simulation avec visualisation en direct obtenue avec matplotlib, il est possible d'utiliser dans le code la partie générant les images dans un sous-dossier Images, et les quelques lignes commentées en bleu pour générer la vidéo directement depuis Python avec ffmpeg.

- La méthode d'Euler est programmée sans utiliser le vecteur  $(x, y, v_x, v_y)$  et sa dérivée  $(v_x, v_y, a_x, a_y)$ . Cela simplifie un peu la donne. On met à jour les vitesses par les accélérations, puis les positions par les vitesses. Il est très simple de faire évoluer le code pour proposer la mise en forme « classique ».