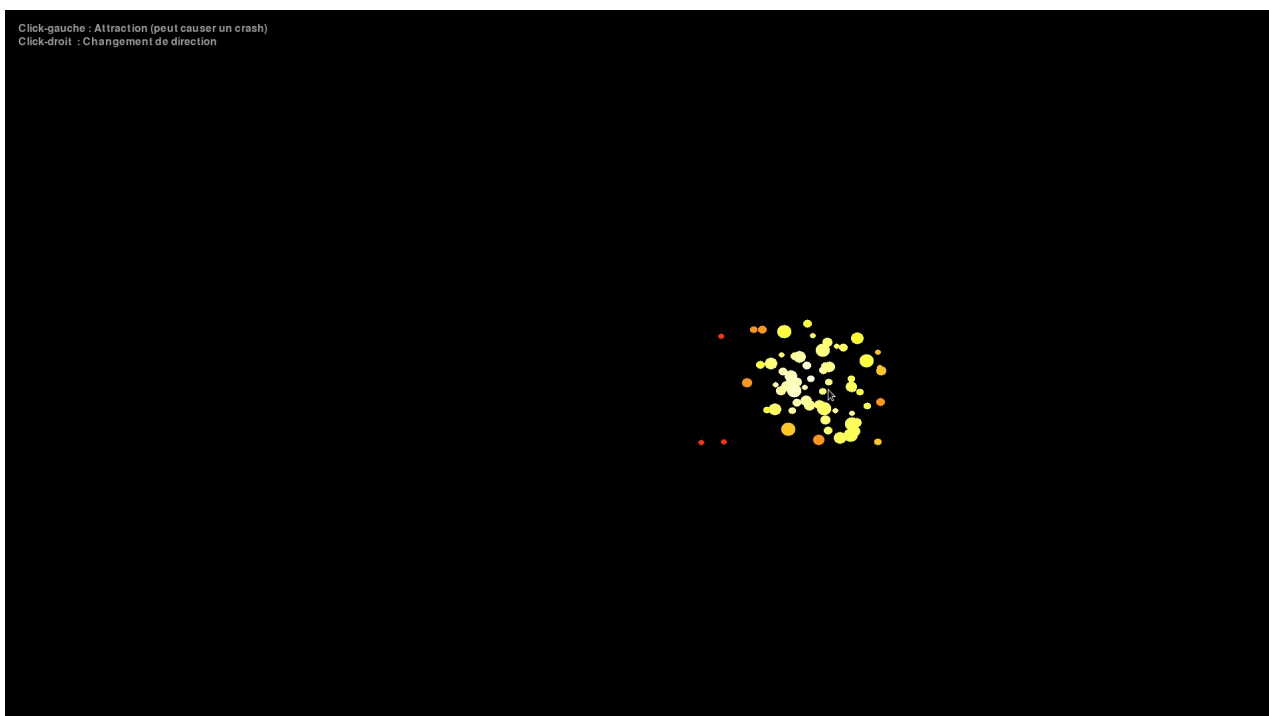
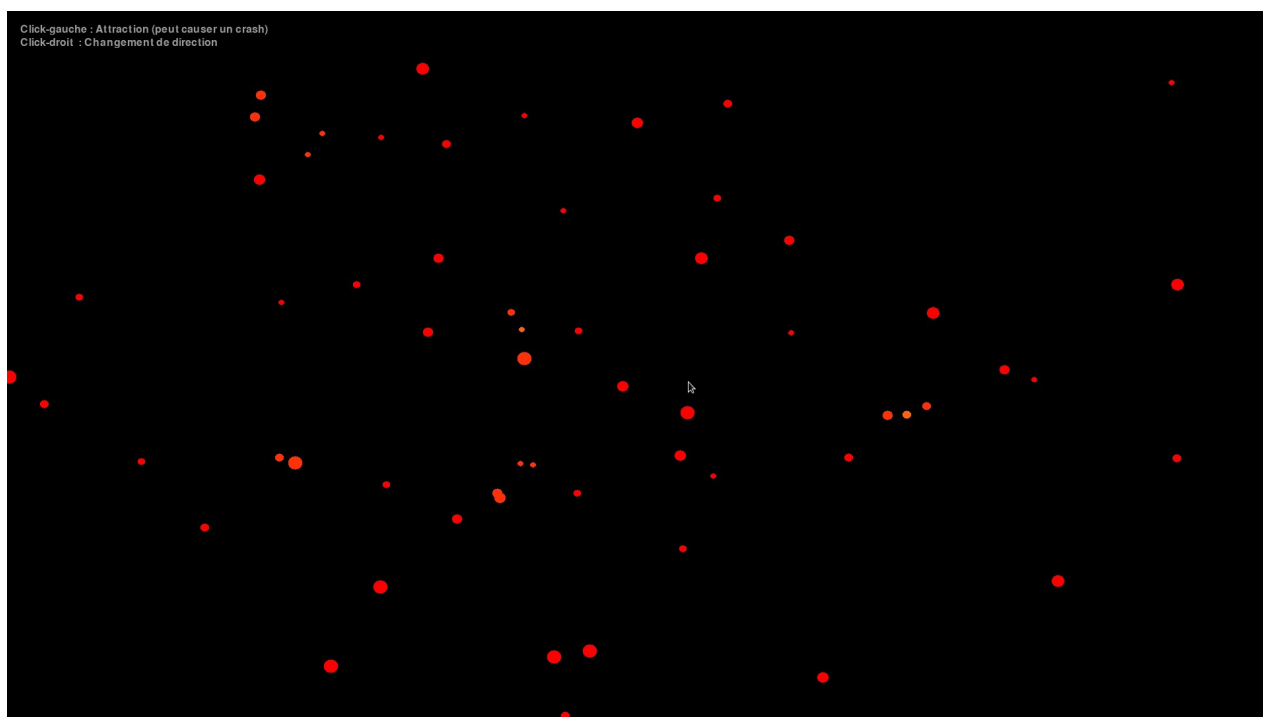


NOM DU PROJET : Simulation d'un système de particules



> PRÉSENTATION GÉNÉRALE :

Ce projet découle d'un ancien projet réalisé en JavaScript. Le but de cet ancien projet était de réaliser un système de particules (en 2 dimensions) qui interagissent entre elles avec un centre de gravité. L'objectif du nouveau projet est de traduire notre précédent programme vers Python et d'apporter de nouvelles fonctionnalités.

Étant des élèves d'un lycée scientifique, l'intérêt de ce projet était d'utiliser ce côté scientifique appris durant nos 3 années au lycée (ici les maths et la physique).

Dans ce projet, nous avons utilisé la programmation en Python apprise en NSI et la programmation orientée objet (assez pratique pour définir une particule).

Pour résumer, quand le script python est exécuté, une fenêtre s'ouvre et affiche le système de particules souhaité (nombre de particules choisit). On peut ensuite interagir avec ces particules avec la souris, soit en la bougeant, soit en utilisant le clic-gauche/clic-droit.

> ORGANISATION DU TRAVAIL :

Nous sommes 3 élèves de Terminale Général, spécialité NSI : Axel, Amine et Maxime.

Axel s'est principalement occupé de l'implémentation de la gravité et du mouvement des particules, plus l'interaction avec la souris.

Maxime a codé le changement de couleur et a écrit la documentation/docstrings.

Amine s'est occupé de la fenêtre d'affichage et la présentation de notre projet.

Nous travaillions au moins 2h par semaine en groupe sur ce projet, plus des séances personnelles faites à la maison (typiquement de la correction de bugs).

Chacun utilisait son éditeur de code préféré, le partage du code source se réalisait sur un dépôt GitHub.

Nous communiquions principalement sur un groupe Instagram.

LES ÉTAPES DU PROJET :

31/03 durée : 2h

- Recherche collective d'idées de projet intéressant
- idée choisie : traduire et améliorer un ancien projet (initialement en javascript via la bibliothèque p5.js)
- Traduction en Python commencée

04/04 durée : 1h

- Idées des futures fonctionnalités
- Répartition du travail
- Finalisation de la traduction en Python

07/04 durée : 1h30

- Axel → attraction autour de la souris
- Maxime → refonte de couleurs
- Amine → définition des bords

13/04 durée : 2h

- Axel → actions des boutons de la souris
- Amine → commence la présentation du projet
- Maxime → docstrings + documentation

19/04 durée : séances personnelles

- Maxime → docstrings + documentation finies (avec Sphinx)
- Amine → finalisation de la présentation du projet
- Axel → dossier de candidature créé et finit

> FONCTIONNEMENT ET OPÉRATIONNALITÉ :

A ce jour, le projet est terminé. Nos objectifs principaux ont été atteints avec succès.

La fin de ce projet fut principalement une phase de déboguage.

Pour cela, nous avons exécuté le programme sur Linux/Windows et interagi avec (souris), puis nous avons corrigé les bugs aperçus.

Certains bugs restent présent dans notre dernière version :

- Certaines particules sortent de la fenêtre avec leur vitesse (ce qui ne devrait techniquement pas arrivé) et leurs coordonnées deviennent tellement grandes que programme plante (dépassement d'entier)
- A bout d'un certain temps (1-2 min) le programme peut planter (même cause que précédemment)
- Le clic-gauche peut provoquer ce bug également (mais ici le bug provient de la distance très petite entre les particules)

> OUVERTURE :

Quelques idées d'amélioration peuvent être envisagées dans le futur:

- Passer en 3 dimensions
- Se rapprocher d'une simulation de galaxie où les particules deviennent des étoiles et le centre de gravité devient un trou noir
- Utiliser l'algorithme de Barnes-Hut pour simplifier la simulation et gagner en performances (l'algorithme considère un groupe de particules proches comme une seule particule dans les calculs de mouvements)

Notre projet peut être accessible au public via un dépôt github disponible avec l'URL :

https://github.com/MaxiLan/trophee_nsi

Ou le QR code :



DOCUMENTATION

Langage de programmation utilisé : Python 3.10

Bibliothèques utilisées : pygame 2.3 (à télécharger préalablement) / random

Le code étant écrit en Python, il est compatible sur la vaste majorité des architectures.

À partir du dépôt GitHub https://github.com/MaxiLan/trophee_nsi:

- Pour démarrer la simulation, il faut exécuter le fichier Python *script.py* situé dans le dossier dossier technique/sources/ (le fichier *particule.py* est nécessaire à l'exécution !)
- La documentation du projet se trouve sur un site web accessible via le fichier *index.html* situé dans le dossier dossier technique/doc/ (le dossier ressources_doc est nécessaire pour utiliser la documentation)
- Le guide d'utilisation se trouve dans le dossier dossier technique sous le nom *protocole_utilisation.md* Vous y trouverez les pré-requis et un guide pour lancer une simulation sur Windows/Linux.