L2 - Projet de Conception Objet

Développement d'un Space Invaders en python

Mettez vous en binôme et veuillez bien lire l'intégralité de ce document avant de commencer

Plan de travail

Le projet comprend 3 parties:

1. Travail sur les exercices préliminaires
2. Développement de la version minimale
3. Enrichissements pour se rapprocher de la définition originale du jeux

1 Exercices préliminaires

Les fichiers qu'il vous est demandé de récupérer sont sur le moodle.

1.1 Exercice 1

Fichier *_Exercice1.py.*

La classe *Example* permet d'ouvrir une fenêtre avec un *Canvas*. Compléter la fonction *install_in* pour afficher un carré rouge de 50 points de coté en x=0 et y=0 (coin haut gauche).

1.2 Exercice 2

Fichier *_Exercice2.py.*

Modifier les fonctions *install_in* et *keypress* pour dessiner un carré rouge au centre et déplacer le carré rouge de 10 points vers la gauche quand on appuie sur la flèche gauche, de 10 points vers la droite avec la flèche droite et de 10 points vers le haut avec la touche *Espace.*

1.3 Exercice 3

Fichier *_Exercice3.py.*

Modifier les fonction *install* et *animation* pour animer un carré rouge de la façon suivante :

- le rectangle est créé en x = 0 et y = 0
- toutes les 300ms, il avance de 20 points horizontalement
- si le x du coté droit du carré rouge dépasse la largeur du canvas, alors le carré retourne en x = 0

1.4 Exercice 4

Fichier *_Exercice4.py.*

Reprendre l'exercice 3 mais cette fois en construisant une classe *Mobile* pour modéliser le carré qui se déplace.

1.5 Exercice 5

Fichier *_Exercice5.py.*

Reprendre l'exercice 4 mais cette fois en construisant une classe *MobileFleet* pour modéliser une flotte de plusieurs instances de la classe *Mobile*.

Chaque *Mobile* a en plus sa propre vitesse de déplacement de sorte que l'animation montre plusieurs *Mobile* qui se déplacent horizontalement comme précédemment mais a des vitesses différentes.

2 Etape 2

Vous développez la version présentée en cours avec les classes *Alien*,

Fleet, *Defender*, *Bullet*, *Game* et *SpaceInvaders* (attention, ces classes sont obligatoires). Vous développez progressivement. Suivez la feuille de route décrite ci-après.

2.1 Fenêtre principale

La fenêtre principale s'ouvre avec une Frame contenant un Canvas.

2.2 Introduction du Defender

Le *Defender* est dessiné et peut être déplacé de gauche à droite avec les touches flèches gauche et droite du clavier

2.3 Le Defender est capable de tirer

Lorsque la touche espace est appuyée, une instance de *Bullet* est crée et représentée par un cercle au dessus du *Defender*. Les cercles des *Bullet* sont animés vers le haut de l'écran.
Un *Defender* ne peut tirer que des rafales de 8 *Bullet* au maximum.
Lorsqu'un *Bullet* parvient en haut de l'écran alors le *Defender* retrouve la possibilité de tirer une nouvelle fois (recharge de 1 coup).

2.4 Flotte des Alien

La classe *Fleet* permet de gérer une flotte *d'Alien*. Les *Alien* sont dessinés dans une matrice. Cette matrice se déplace de gauche à droite et de droite à gauche. Elle avance vers le bas à chaque fois qu'elle atteint un des deux bords. Si l'ordonnée du bord bas de la matrice des *Alien* est supérieure à l'ordonnée du bord haut du *Defender* alors la partie est perdue.

2.5 Collisions Bullet/Alien

Lorsqu'un *Bullet* touche un Alien alors l'Alien qui est touché n'est plus dessiné. Le *Defender* dispose d'une recharge de 1 coup.
Une explosion peut être dessinée pendant quelques millisecondes quand un *Bullet* touche un Alien. Si tous les *Alien* sont touchés alors la partie est gagnée.

3 Etape 3 Version améliorée du jeu

Vous programmez d'abord obligatoirement la sauvegarde des scores. Vous appliquerez ainsi le complément de cours sur les fichiers qui vous sera présenté en CM pendant les semaines du projet.

3.1 Sauvegarde des scores

Vous mettez en place le comptage et la sauvegarde des scores dans un fichier. On a donc une gestion des joueurs avec la persistance pour l'inscription de joueurs et l'enregistrement des scores dans un fichier. Un panneau introductif est affiché au démarrage du jeu. Sur ce panneau :

- La liste des scores sauvegardés est affiché avec le nom des joueurs et leur score ;
- Le joueur doit inscrire ou choisir une ligne dans la liste;
- Le nouveau score du joueur est calculé en cours de partie ;
- En fin de partie, les scores sont mis à jours dans le fichier de sauvegarde avec celui du joueur qui vient de terminer sa partie.

Sur le panneau principal du jeu, une barre présentant le score courant est affiché.

3.2 Autres améliorations

Vous ajoutez des éléments qui améliorent le jeux. Par exemple :

- **Plusieurs vies disponibles**: le Defender a donc plusieurs vies, les vies restantes sont affichées sur la barre des scores ;
- **Les Alien peuvent tirer eux aussi** : le tir d'un alien peut toucher le Defender qui pert alors une vie.
- **Le Defender peut s'abriter derrière des bunkers de protection** : le bunker peut être progressivement détruit par les tirs des Alien.

- **Ajout d'animations** : par exemple, les Alien peuvent être animés pendant leur déplacement ou changer de forme en fonction de leur état
- **Autres compléments ...**

4 Organisation

Vous avez 8 heures encadrées pour le développement du projet : vous avez 3 séances de TP et 1 séance de TD. Il vous faudra certainement compléter avec du travail personnel.

La présence est obligatoire pour tous pendant les heures de TP et TD prévues.

4.1 Comment développer

4.1.1 Travail en binôme

Vous travaillez en binôme mais sur une et une seule machine. Donc on a toujours un premier développeur qui utilise le clavier et la souris et le second développeur qui lit, est vigilent sur ce qu'écrit le premier, commente oralement réfléchit avec plus de recul sur ce qui est en cours.

Inversez les rôles, le clavier doit changer de main régulièrement.

4.1.2 Environnement de développement

Concernant l'édition des programmes et leur exécution, vous utilisez l'environnement des TD/TP. L'utilisation d'un autre éditeur est autorisé mais sans aucun support de la part de l'enseignant.

4.1.3 Composants logiciels importés

Seul *TkInter* est autorisé pour l'étape 1 (pas de PyGame par exemple). Pour l'étape 2, veuillez demander à l'enseignant si vous envisagez d'utiliser un import pour des composants ou des packages particuliers.

4.2 Ce que vous devez rendre

Ce que vous rendez comprend le code python accompagné d'une documentation. Les noms des membres du binôme doit être indiqué en entête de chaque source et dans votre documentation.

La documentation décrit :

- les différentes classes avec leur structure et leurs responsabilités fonctionnelles
- les relations entre les différents objets du système
- les particularités que vous souhaitez mettre en valeur et qui ne sont par facilement visibles par la simple lecture du code

Il est demandé de conclure avec des points critiques sur votre développement : ce que vous avez fait et ce que vous n'avez pas fait et/ou des améliorations possibles/souhaitables du codage et le pourquoi de ses améliorations.

4.3 Comment rendre votre projet

Vous devez rendre votre projet en utilisant le moodle.

4.4 Démonstrations

Vous devrez faire une démonstration pour montrer votre version finale pendant la dernière séance de TP

5 Ce qu'il faut faire pour ne pas réussir votre projet

Pour ne pas réussir votre projet, suivez les conseils suivants :

- votre modèle de l'étape 1 ne comprend pas les classes demandées
- vos classes ont 40 variables d'instances

- vos fonctions font au minimum 50 lignes
- le niveau d'imbrication moyen est de 6
- vous utilisez des variable globale et des fonctions en dehors des classes
- le code n’est pas commenté
- les algorithmes complexes ne sont pas commentés
- vos noms de fonctions et de variables ne sont pas judicieusement choisis
- vous avez des variables ou des fonctions non utilisées
- vous n’utilisez pas les accesseurs pour accéder aux propriétés des objets
- vous.avez().des().appels().a().de().fonctions().incompréhensibles() .et().a().rallonge()
- vous ne gardez aucune version intermédiaire qui marche au fur et à mesure que vous avancez.
- au final, ça ne compile pas et on ne peut rien faire tourner.