

OC PROJET 03 – Aidez MacGyver à s'échapper !

1. But du projet

Dans ce projet, le but est de créer un jeu 2D avec Pygame contenant un labyrinthe (les murs), deux personnages (MacGyver et le Gardien) et 3 éléments (Aiguille, Ether et Tube plastique) qui en formeront un quatrième (la seringue).

MacGyver doit ramasser tous les objets avant de se diriger vers le gardien pour l'endormir, s'il ne les a pas tous ramassés, la partie s'arrête et le joueur a perdu.

2. Contraintes du projet

1. Il n'y a qu'un parcours.
2. Un compteur doit lister les objets
3. Les objets sont placés aléatoirement à chaque fois que le jeu est lancé
4. MacGyver ne se déplace qu'avec les flèches directionnelles
5. Le labyrinthe doit être dans un fichier séparé pour pouvoir le modifier facilement
6. Un fichier par classe
7. Une fonction main
8. Respect de la PEP8

J'ai donc créé un fichier :

- **"constants"** pour la gestion des constantes du jeu
- **"maze"** pour la gestion du labyrinthe
- **"player"** pour la gestion de MacGyver et ses mouvements et pour le ramassage des objets
- **"itemcounter"** pour la gestion du compteur
- **"initializer"** qui est le fichier principal
- **"main.py"** qui lance le jeu

3. La classe GameBoard (Dans maze.py):

Ici, le fond d'écran est en fait un mur dans lequel je vais venir dessiner le chemin sur lequel MacGyver va pouvoir se déplacer.

Je lis le fichier de structure du labyrinthe où les 0 correspondent au chemin possible, le G est le gardien et les 1 sont les murs.

Je vais donc lire les 0 et remplacer les murs par le fond d'écran. Je vais chercher le gardien et créer une liste contenant les 3 objets que MacGyver doit ramasser.

Ensuite, je place et affiche tous les éléments sur le plateau. J'utilise la fonction **"sample"** du module **"random"** pour le placement aléatoire des objets. Ici, ce sont les images qui sont choisies aléatoirement dans la liste.

Enfin, je crée une liste vide dans laquelle je vais ajouter chaque association **"image / position"** après que le programme les a positionnées sur le plateau.

Je crée une fonction de vérification du compteur et de récupération des objets ramassés pour les placer dans la barre d'éléments

Je crée une fonction pour la fin de partie qui vérifie la position de MacGyver par rapport au gardien puis la valeur du compteur. S'il est égal à 0, la partie est gagnée, sinon elle est perdue.

4. La classe Player :

Cette classe gère MacGyver, ses mouvements et le ramassage des objets.

Je vais initialiser son image et sa position de départ sur le plateau puis les offsets de chaque touche directionnelle.

Chaque mouvement se décompose en 3 phases :

1. **L'effacement de MacGyver** : où son image est remplacée par une une partie du fond d'écran
2. **La mise à jour de sa position** : où l'emplacement choisi est vérifié, libre, dans la liste des déplacements possibles dans la structure du labyrinthe.
3. **L'affiche de MacGyver** à sa nouvelles position.

Ensuite, je crée la fonction de ramassage des objets, qui va utiliser la liste des associations "**image / position**" pour afficher la bonne image dans la barre d'éléments une fois ramassée et effacée de la liste des objets puis je décrémente le compteur à chaque objet ramassé.

6. La classe ItemCounter :

Fichier de gestion du compteur, initialisé à 3 au début de la partie. Une fonction se charge de la décrémentation et l'autre retourne la valeur du compteur, mis à jour à chaque tour de boucle.

5. Le fichier initializer :

C'est le fichier principal. Il est optimisé au maximum pour ne gérer que les appels de fonctions et la gestion des touches de sortie du programme.

J'ai créé une fonction "**main**", elle-même appelée par un fichier "**main.py**".

6. Le fichier constants :

C'est le fichier qui contient toutes les constantes du jeu :

- La taille de la fenêtre
- Les images
- La taille et le nombre de sprites
- le titre de la fenêtre, etc.

[Lien GitHub](#)