

Macgyver Projet 3

<https://github.com/hakan7822/Projet3Macgyver>

I Mon raisonnement:

Pour réaliser ce jeu j'ai tout d'abord cherché à ouvrir une fenêtre vierge avec Pygame. Une fois cette étape réalisée, j'ai voulu créer un fichier texte que j'ai par la suite généré sous forme d'affichage graphique en lisant chaque lettre une par une .

Jusqu'ici ça ne m'a pas pris beaucoup de temps. Dans mon raisonnement, il me manquait le fait d'afficher les différents éléments de mon labyrinthe et leurs caractéristiques. Avec surtout, les items affichés dans des positions aléatoires. Ainsi, après cela je pouvais arriver à une étape où tout les éléments de mon labyrinthe pouvaient être affichés mais sans avoir tout les principes du jeu. MacGyver pouvait bouger tout en évitant les murs.

Le programme vérifiait chaque case avant de s'y rendre. Ainsi, si c'était un mur le mouvement était nul. Pour en arriver là j'ai effectuée des captures d'événements. MacGyver pouvait passer sur les objets sans pour autant les ramasser et arriver devant le gardien mais rien ne se passait.

Il ne me manquait plus qu'à ramasser les objets dont MacGyver égalerait les coordonnées et terminer la partie sur une victoire, si tout est ramassé ou sur une défaite si rien ou une partie est ramassée.

Pour conclure, la difficulté principale pour moi était la gestion de la création du jeu et surtout la gestion pour faire bouger MacGyver tout en indiquant qu'il fallait ramasser les objets et gagner.

N'ayant pas d'expérience dans ce domaine, ce n'était pas facile de penser à l'avance aux boucles, aux conditions, ainsi qu'à l'exécution de certains événement. Par exemple, pour donner de la mobilité au personnage ce n'est pas de simple mouvement mais une gestion de tout erreur à éviter (*exemple: éviter les murs, ramasser les objets*) et à penser au rafraîchissement de la fenêtre pour chaque nouvelle affichage. Je crois donc, qu'il faut apprendre à penser ainsi pour coder en python.

II Mes fichiers:

Avant de me lancer, il fallait que je me facilite la tâche au plus possible, tout en ayant une bonne organisation, ainsi que de la clarté dans mon programme. J'ai donc, en dehors du fichier texte créé neuf fichiers Python comprenant trois catégories :

● Les Fichiers classes :

J'en ai créé quatre, ces fichiers classes m'ont servi à définir MacGyver, le Gardien, les Items et les cases (Square). Chaque éléments du jeu est défini dans un fichier

représenté par une classe avec des fonctions qui servent dans divers étapes de la création du jeu.

- **Les Fichiers méthodes :**

J'ai crée trois fichiers méthodes :

- `Generate_laby()` : Génère le labyrinthe en lisant le fichier texte.
- `Display_laby()` : Affiche le labyrinthe.
- `Put_item()` : Attribut un set de coordonnées aléatoire à l'objet.

- **Le Fichier d'exécution :**

Ce fichier, que j'ai nommé `Labyrinthe.py` lance le jeu, elle définit les différents items. Elle contient deux boucles, dont la première sert à faire tourner le jeu et la deuxième sert à faire des captures de chaque événements du jeu. Elle contient, pour terminer, l'ensemble des mouvements de MacGyver (avec les touches du clavier), qui sont soumis aux divers conditions. Ainsi, s'il passe sur les objets, ils les ramassent et s'il a ramassé tout les objets il gagne, sinon il perd.

Précision:

Les classe "MacGyver" et "Exit" sont identiques, pour faciliter l'organisation du programme j'ai opté pour la création de deux classes.

J'ai crée les Coordonnées sous forme de classe car je trouve cette méthode plus pratique.