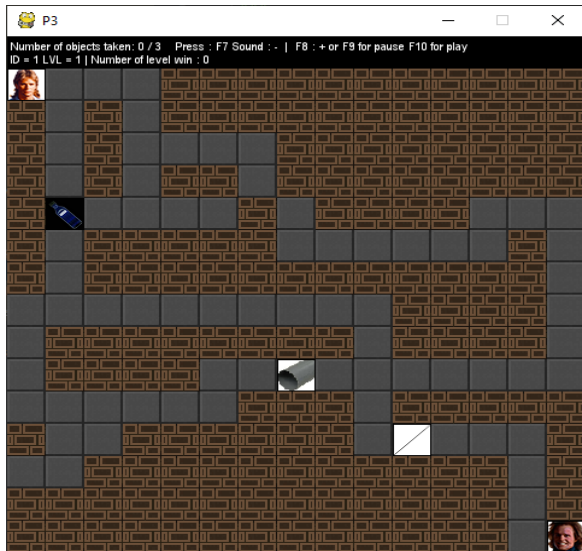


Projet 3 : Aider MacGyver à s'échapper !

<https://github.com/Anthony10700/Project3>

Dans ce projet on nous a demandé de créer un jeu avec la bibliothèque PyGame, le but étant dans un labyrinthe de ramasser des objets au nombre de trois et de se présenter à la sortie où un gardien attend. Si l'on se présente devant le gardien sans la totalité des objets nous mourrons.



Ma démarche :

J'ai déjà commencé par m'imaginer tous ce que je devais faire, j'ai pris note de chaque tâches :

- Lire la doc de PyGame.
- Penser la structure du jeu.
- Différencier un peu près les tâches de mon code
- Effectuer l'algorithme du jeu en schéma (voir ci-dessous).

Concernant mes problèmes rencontrés :

Je n'ai pas rencontré spécialement de problème dans ce qui était demandé. J'ai effectué une fonction en plus de tous ce qui était demandé pour générer une carte aléatoirement, j'ai eu beaucoup de mal au démarrage et je l'ai recommencé plusieurs fois. Pour réussir je vais étape par étape et effectue des simulations pour voir si mon idée peut fonctionner.

Mon plus gros problème si on peut voir ceci comme un problème a été que ma première version de mon code était trop procédurale d'après mon mentor. Comme je suis ancien dev en VB .net et VBA, la POO était pour moi assez abstraite. J'ai dû donc revoir tous mon code mais le plus dur avait déjà été fait, la POO c'est vraiment une façon de voir les choses.

Mon projet se décompose comme suivant :

- Fichier main.py, contient la boucle de jeu et les instances de class utiles.
- Fichier constant.py, contient toutes les constantes utiles à l'ensemble du programme.
- Fichier gamepy.py, contient toutes les méthodes pour le contenu du jeu, liste de la MAP, ID, LVL, nombre de parties gagnées etc.
- Fichier display.py, contient tout ce qui concerne l'affichage de PyGame et les méthodes d'affichage de MAP, ou de message.
- Fichier macgyver.py, contient la class pour l'objet macgyver qui sera créé pour le jeu, sa position en sprite, ses méthodes : déplacements possibles etc.
- Fichier objetinmap.py, contient la class pour créer un objet à déposer sur la map, il a ses propres méthodes comme positionner aléatoirement l'objet etc.
- Un fichier map.json contenant l'objet map lu par l'objet gamepy.
- Le fichier requirements.txt, qui contient les modules complémentaires à installer.
- Le dossier ressource, contenant toutes les images nécessaires au jeu.

L'algorithme :

