

WALL IS YOU

Projet informatique 2

L1 Mathématiques et Informatique

Université Gustave Eiffel

Année 2022-2023 – Semestre 2

Antony Navarro

Karl Moukheiber

Groupe Projet TP3_10



1. Manuel utilisateur

- Lancement du programme :

Pour lancer le programme il suffit de lancer "Run" sur le fichier "wallisyou.py" sur VSCode par exemple et c'est suffisant pour nous envoyer sur la fenêtre fltk et commencer à jouer.

Ou bien depuis la console en se plaçant dans le répertoire Projet2 la ligne de commande `python3 wallisyou.py` permet de lancer le jeu.

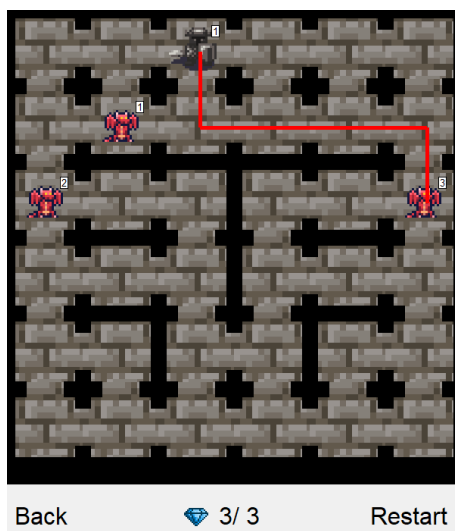
- Paramètres sur la ligne de commande :

Nous n'avons pas rajouté de paramètres sur la ligne de commande. Tout peut se paramétrer graphiquement depuis le jeu.

- Navigation dans les menus :

Pour retourner au menu il faut appuyer sur le bouton « retour » ou sur la touche « escape »

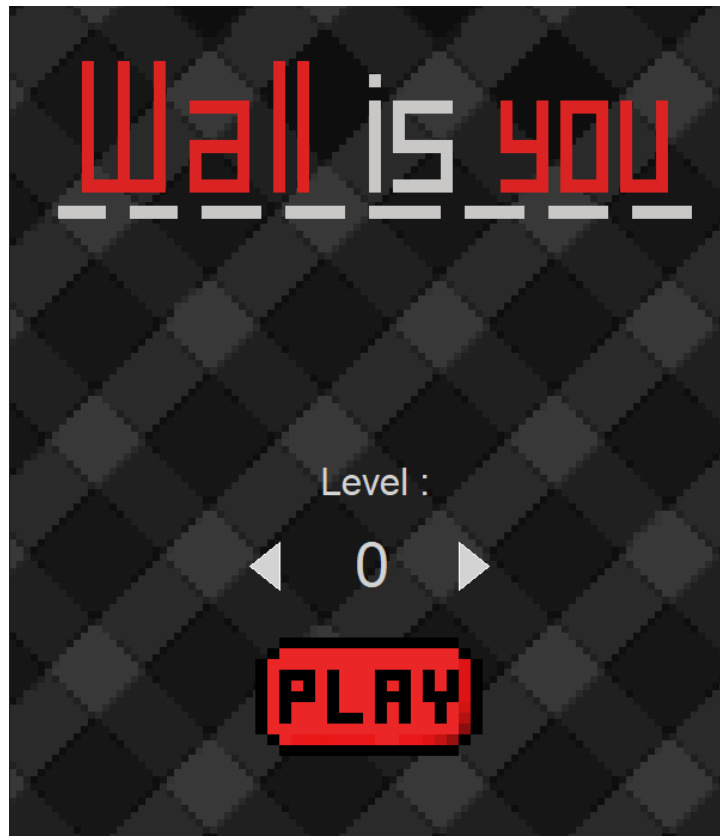
Pour relancer le jeu il faut appuyer sur « restart » ou sur ma touche « r »



- Lancement du jeu :

Pour choisir le niveau il suffit de se servir des flèches, soit depuis le clavier ou bien en cliquant dessus.

Pour lancer le jeu il est possible de cliquer sur le bouton « Play » ou bien juste d'appuyer sur « enter ».



- Commandes à la souris et au clavier :

- Pour pivoter une case : Clic gauche sur cette case pour qu'elle fasse une rotation vers la droite
- Pour poser un trésor sur la case souhaitée : Clic droit sur cette case
- Pour déplacer le joueur à la fin du tour : « espace »
- Pour relancer la partie : Clic gauche sur "Restart" ou la touche « r »
- Pour retourner au menu : Clic gauche sur "Back" ou la touche « escape »
- Pour choisir le niveau : Clic gauche sur les flèches (on peut également se servir des flèches du clavier)

2. Etat d'avancement du projet

- Fonctionnalités réalisées et non réalisées :

Toutes les fonctions obligatoires ont été réalisées. En ce qui est des fonctions optionnelles nous avons rajouté les trésors, le temps et le nombre de rotations et plusieurs améliorations d'affichage (menu, logo, arrière-plan, confettis etc.).

- Problèmes connus :

Pas de problèmes connus, aucun bug n'a pu être trouvé.

- Améliorations réalisées :

En ce qui concerne le travail optionnel nous avons rajouté le chemin le plus court, les trésors, le temps et le nombre de moves, les différents records réalisés à chaque partie et plusieurs améliorations d'affichage (menu, logo, arrière-plan, confettis etc.).

- Explications des choix, en particulier :

- Résolution astucieuse de problèmes difficiles

L'utilisation de class python a rendu l'avancement du projet beaucoup plus simple et le code plus facile à lire.

- Explications de fonctions complexes

La fonction intention (récursive) :

La fonction intention utilise une approche récursive pour trouver le chemin le plus court menant au dragon de plus haut niveau dans un donjon. Elle explore les positions valides et connectées à partir de la position actuelle, en mettant à jour le meilleur chemin et le niveau du dragon accessible à chaque itération. Si aucun chemin n'est trouvé, elle renvoie None. Sinon, elle renvoie le meilleur chemin trouvé jusqu'au dragon de plus haut niveau.

La fonction intention (chemin le plus court) :

La fonction intention utilise une approche de parcours en largeur pour trouver le chemin le plus court menant au dragon de plus haut niveau dans un donjon. Elle utilise une file (deque) pour stocker les positions à explorer et les chemins parcourus jusqu'à ces positions.

Elle commence par initialiser une file avec la position de départ et une liste vide représentant le chemin initial. Ensuite, elle utilise une boucle principale pour explorer les positions de manière itérative.

À chaque itération, elle extrait la position et le chemin correspondant de la file. Si la position correspond à un dragon et que son niveau est supérieur au niveau du dragon de plus haut niveau trouvé jusqu'à présent, elle met à jour le niveau maximal et le chemin le plus court.

Elle explore ensuite les positions adjacentes valides à partir de la position actuelle, en les ajoutant à la file avec le chemin mis à jour. Les positions déjà visitées sont ignorées pour éviter les boucles infinies.

Une fois que la file est vide, la fonction renvoie le chemin le plus court trouvé jusqu'au dragon de plus haut niveau. Si aucun chemin n'a été trouvé, elle renvoie une liste vide.

3. Organisation du travail

- Répartition quantifiée des tâches :

Karl : Fonctions pivoter, connecter, rencontre, fin_partie, affichage des records + compte rendu du projet.

Antony : Boucle principale du jeu et du menu, Class Aventurier, Dragon et Game, fonctions d'ouverture des fichiers et affichage des donjons, fonction intention récursive, toutes les fonctions d'affichage graphique avec fltk et création des images pour le menu et le jeu, améliorations optionnelles comme le chemin le plus court et les trésors, Docstring pour toutes les fonctions et contribution notable pour le rapport.

La communication du groupe était assez bonne cependant Karl a rencontré plusieurs soucis avec son ordinateur durant les vacances qui n'est pas en bon état et donc le travail durant cette période-là a été moins productive que prévu.

- Difficultés rencontrées :

Avant les vacances il était plus facile d'avancer sur le projet ensemble grâce aux séances de TD mais malheureusement Karl a rencontré des problèmes avec son ordinateur qui lui ont rendu impossible de pouvoir exécuter le programme dessus et pouvait alors travailler uniquement en allant à la bibliothèque Universitaire. Donc pendant les vacances l'avancement du projet n'a pas plus été égal, ce qui explique que la majeure partie du projet a été réalisé par Antony.

4. Conclusion

Karl : Projet sympathique et intéressant qui était plaisant à réaliser. Rien ne nous a déplu.

Antony : J'ai adoré travailler sur ce projet et voir le jeu prendre forme. Je trouve cependant que le premier projet était plus compliqué à réaliser et les mécaniques de jeu étaient également plus intéressantes. Avec un peu plus de temps j'aurais bien aimé pour ajouter plus de fonctionnalité au jeu pour le rendre plus intéressant et amusant à jouer.