

Cahier des charges :

Le but de notre programme est de créer un labyrinthe aléatoirement. Puis afficher un personnage capable de se déplacer en son sein. Les programmes sont réalisés en POO, pour simplifier la lecture du programme et sa construction (plus simple pour la création d'un jeu).

Recherche documentaire :

Pour la réalisation du jeu, nous avons utilisé les sites suivants :

<https://youtu.be/K7vaT8bZRuk>

https://en.wikipedia.org/wiki/Maze_generation_algorithm

Pour l'interface du jeu, nous avons recherché dans des vieux jeux d'arcade pour sortir un concept plutôt simple. Notre attention s'est portée sur le jeu « Gauntlet II » un jeu d'aventure, d'exploration de labyrinthes, qui regorge de monstres et d'objets.



Répartition du travail :

Lors de la réalisation du projet, nous avons fait plusieurs appels, afin de se mettre d'accord sur le projet, les éléments à traiter, etc... Pour la programmation, Allan a produit une majeure partie du code, il m'a bien aidé de mon côté. Pour ma part, je me suis plus focalisé sur l'interface du jeu, des personnages, au décor (texturing). La collaboration s'est faite toute seule, étant tous deux passionnés par les jeux vidéo, l'idée fut très vite claire. De plus, la conception d'un autre jeu dans le passé, nous a simplifié la répartition des tâches car nous étions habitués par le travail d'équipe.

Réalisation des fonctions :

- main.py : permet de lancer le jeu
- map.py : permet de définir la texture des murs, des sols et du labyrinthe
- maze.py : permet la création d'une grille, qui va être trouée à chaque tour de boucle, jusqu'à avoir un labyrinthe complet
- game.py : permet d'update le timer, déplace le joueur, modifie la route du monstre, vérifie si le joueur a fini le labyrinthe, vérifie si le joueur n'est pas dans un mur. Affiche les murs, le sol, le joueur, le monstre et la fin du labyrinthe.
- monstre.py :
- player.py : permet de déplacer le personnage grâce à un rafraîchissement de l'image, répertorie les sprites du personnage.

Fonctionnement :

Le programme fonctionne correctement, puisqu'il se lance lors de l'exécution de la fonction « main.py ».

Extension :

Avec la petite expérience que nous avons dans le développement de jeu, nous pensons qu'il serait bien d'ajouter une personnalisation du personnage, type : différents accessoires, couleur de cheveux, yeux, peau, vêtements etc... De plus, ajouter un menu lors du lancement du jeu.

Conclusion :

Ce projet de jeu vidéo arcade, nous a fait gagner en expérience dans le domaine. Grâce à la conception de chaque programme, de chaque texture, nous avons progressé. Le projet nous a été bénéfique, pour ma part, j'ai progressé en programmation et pour Allan, il aime programmer.

Ce projet a été appris plus comme un plaisir, un loisir, qu'un réel travail. Allan a même créé un algorithme.

PS : UBISOFT a vendu toutes les parts d' Assassin's Creed pour acheter notre Oeuvre et souhaite le commercialiser au plus vite, pour que le monde entier, profite de cette exclusivité bientôt disponible dans tous les magasins.

Bonus :

Voici quelques réalisations de l'interface qui ne sont pas toutes visibles :

