Rapport

Projet POA

Equipe du projet :
YANG Yu
ZHU Tingting

1.Structure des fichiers

1. Labyrinthe

Fonctions : Chargement de la carte, initialiser data.

2. Gardien

Fonctions : Fire, Move, Analyzer les situations différentes et changer le mode，Orientation automatique

3. Chasseur

Fonctions : Juger le résultat.

2.Fonctions réalisées

1.Chargement de la carte

On lit un fichier .txt juste comme l'exemple et charger les charactères un par un. On met '1' comme les murs normaux et les images. '+' pour les corners, on utilise après pour orientation. 'G' pour les gardiens et 'T' pour le trésor.

2. Orientation automatique

On choisit l'algo dijkstra pour l'orientation. On n'a pas parcouru tous les

points de la carte. On choisit les quatre points autour des boxes, corners de mur et trésor comme une nouvelle carte. S'il n'y a pas de barrière entre les deux points on regarde il y a un chemin entre les deux points. Ça va donner le même résultat comme parcourir tous les points de la carte. Parce que la ligne droite est toujours le plus proche entre les deux points.

3 Modes différents

(1) Mode normal

Mode normal est les gardiens ne mouve pas. Normalement ce n'est pas possible d'être mode normal. Je vais l'expliquerai dans les problèmes rencontrés.

(2) Mode attaque

Les gardiens vont chasser le chasseur et le fire quand il peut regarder le chasseur ou le chasseur est très proche au trésor. C'est-à-dire le dernier gardien peut regarder le chasseur.

(3)Mode Patrouiller

Pas vraiment réaliser. Je l'expliquerais après.

4. Juger le résultat

Les gardiens et chasseur ont 100 hp. Pour chaque shot, on regarde le _z de fireball et décider le damage. Si le chasseur a 0 hp ou plus bas. Le jeu est perdu. Si le chasseur shot le trésor il va gagner. Si le chasseur est touché par les gardiens ce jeu est perdu aussi.

3. Fonctions non réalisé et problème rencontré

Pour le mode patrouiller, j'ai fait un version gardien avec le mode patrouiller. Le gardien choisit le plus loin point corner et regarde comme destination. Quand il arrive ce point et il va choisit un nouveau point plus loin et pas de barrière. Le problème est quand j'ajoute un (vector<double> _move_cible) comme un paramètre de gardien. Il y a une segmentation fault après le chargement de la carte, même je n'ai pas utilisé ce vector. Quand je change le vector<double> à vector<int>. Il va update le gardien une seule fois, avec aucun problème erreur de code. Je pense que c'est le problème de mémoire. Je n'ai pas trouvé la solution. J'ai essayé mettre ce paramètre dans la classe labyrinthe et change le vector à une liste et je fais de mon mieux à diminuer les paramètres utilisés sans succès.

4. Conclusion

Pendant ce projet moi et Tingting travaillent dans un très bon air. La communication est très efficace et travailler comme une équipe. Pour nous, c++ est un langage pas familier. On append beaucoup sur ce projet

5. Utilisation

Make
./labh -l labyrinthe.txt