

Projet labyrinthe Mac Gyver

version 2

1) définitions des classes

j'ai besoin:

classe "Zone" (terrain)

classe "Character" (Mac Gyver et le gardien)

classe "Objet" (seringue, aiguille et ether)

classe "Decor" (mur)

"Character", "objet" et "élément de décor" ont la même ossature, je crée une classe mère "objet" et 2 classes fille "personnage" et "élément de décor".

"Décor" aura une liste des positions de ses emplacements

"Character" aura un sac pour transporter des objets récupérer, <<move>> pour se déplacer, et 2 fonctions pour récupérer les objets et combattre le gardien

"Zone" aura tous les éléments de gestion du terrain création etc...

2) pour créer la surface du jeu, j'ai besoin d'une carte du labyrinthe, emplacements des personnes

Idéalement c'est de définir à l'avance les lieux où seront des objets pour qu'ils ne soient pas sur le trajet de Mac Gyver

j'ai choisi de faire un outil (pil_image.py dans répertoire "zones")

je dessine dans un BMP(16 colors) le labyrinthe:

le noir => le trajet

le blanc => les murs

le rouge => emplacement éventuel objet

je récupère le niveau de couleur dans la fonction et crée un txt avec les éléments récupérés.

j'aurais pu inclure la fonction a "Zone" mais je sais pas si le niveau trouvé est le même quel que soit le pc ou stocker sous forme de bit ex : 0xff dessine => 11111111 etc.. mais le choix du txt permet une visualisation du rendu

3) je fais l'initialisation avec tous les objets et personnages comme cela on peut customiser le labyrinthe

le terrain fera 15 cases avec un longueur de 25 pixels soit 375

4) on active les events sur 2 touches pour démarrer ou quitter ("echap" et "space")

5) on presse "space" et le labyrinthe s'initialise avec supp de event sur "space" et activation sur "up, down...."

On récupère les positions de tous les éléments dans le fichier txt sous forme x,y.

Pour les objets (seringue...) on tire au hasard 3 emplacements possibles qui ont été définis.

On crée une image de tous les éléments avec le couple => position * nombre de pixel (25)

6) on bouge "Mac Gyver"

la position des éléments sont stockes sous forme x,y avec une valeur 0 à 14

Les 4 events de direction renvoi des valeurs 1,0 ou -1,0 ou 0,1 ou 0,-1

celle-ci est ajouté à la position de Mac Giver et on test si notre couple n'est pas dans la liste des positions des murs, objets, gardien ou en dehors des limites (0 à 14)

7) Si nouvelle position dans liste des mur , on bouge pas ou hors limite.

Si on est sur un objet on, le récupère dans le sac , il passe de la liste au sac avec affichage a l'écran

Si on est sur position gardien on test si on a les 3 objets on gagne sinon on perd

8) on réinitialise les données et on purge la zone de jeu de tous ses éléments

On désactive les mouvements et on réactive le "space" pour relance d'une partie