README

Projet programmation d’interface Sophie Bousquet, Dounia Hullot.

Parties du sujet traité :

-Planification des actions

-Planification du nombre de tours

-Déplacement des joueurs

-Déplacement du Marshall

-Création d’un menu du jeu

-Ajout de la musique

-Dessin du fond, train, personnage, avatar…

Problème rencontré qu’on a réussis à éliminer :

-Faire avancer et reculer le Marshall aléatoire (trois fonctions utilisées : une pour faire avancer le Marshall, une autre pour le faire reculer, et la dernière est une fonction prédéfinie (rand) pour générer des valeurs aléatoires).

-Image sur un bouton qui ne voulait pas s’afficher (on a changé l’image avec une autre).

-Problème de récupération des valeurs des combobox parce qu'on récupérait les actions dans une fonction (recuPaction) mais il fallait mettre la fonction en event. On avait besoin d’utiliser la liste self.action que nous ne pouvions pas appeler dans la fonction,  parce que la fonction est un event et la liste est self.action, donc on a créé la liste action dans un autre fichier qu’on a importé dans le fichier principal.

-Les profils des joueurs ne voulaient pas changer à chaque tour pour montrer quel joueur joue.

Problème rencontré que nous n’avons pas pu éliminer :

-Action tir et action braqué

-Changer l’image de profil des joueur

-Image en Canvas pour la transparence du train parcequ’on c’est mélangé dans les .pack(), .place() et .grid()

-Définition du nombre de tour avec le entry

-Définition du nombre d’actions

-Perdre un butin lorsqu’on croise le Marshall

-Lorsqu’un joueur croise le Marshall il faut cliquer sur action 4 fois pour que les 4 personnages puisse chacun fuir le Marshall.

-Le dimensionnement de la fenêtre, d’un ordinateur à un autre la fenêtre peut être zoomer et faire disparaître certaine chose.

Déplacement du Marshall :

Le Marshall se déplace à chaque fois qu’un des joueurs se déplace, c’est à dire lorsque l’on appuie sur le bouton Action.

Le Marshall se déplace avec une probabilité de 1 chance sur 2 d’avancer ou de reculer.

Pour le fonctionnement du Marshall nous avons crée 3 fonctions, une qui sert à faire avancer le Marshall, une autre pour le faire reculer et la dernière qui sert à le faire se déplacer de manière aléatoire, grâce au 2 fonctions précédente.

La fonction de déplacement du Marshall est appelé dans la fonction action\_exe() afin qu’il se déplace à chaque mouvements des joueurs.

Morceau de code écrits à plusieurs ou emprunter :

Nous avons tout fait seulement à deux.