

Rapport final CHOUKROUN - MITTAIN Projet Python

Jeu des allumettes, à rendre le dimanche 14 janvier avant 22h

I) Notre thème principal :

Notre thème est « les animaux aquatiques » et nous avons représenté les allumettes par des petites tortues pour faire un clin d'œil au module turtle qui nous a permis de construire ce projet.

Notre fond est constitué de 2 poissons colorés avec un fond bleu et des vagues avec l'écume représentées en blanc. Nous avons voulu garder un décor assez simple pour ne pas surcharger la zone d'affichage du jeu et ainsi laisser les écritures du jeu lisibles.

II) Nos règles du jeu :

Les règles du jeu sont définies au début de notre programme principal. Elles s'appuient sur la version de base énoncée dans le projet pédagogique :

Le joueur joue contre l'ordinateur.

Un nombre aléatoire entre 11 et 17 tortues-allumettes est tiré par l'ordinateur. Les tortues-allumettes sont affichées au joueur. Le joueur tire entre 1 et 3 tortue(s)-allumette(s). Le nombre de tortues-allumettes restant est calculé, et celles-ci sont de nouveau affichées. Puis l'ordinateur joue : il retire un nombre aléatoire entre 1 et 3 tortue(s)-allumette(s). Le nombre de tortues-allumettes restant est de nouveau calculé et affiché au joueur. Le joueur joue, l'ordinateur joue,...et ainsi de suite, jusqu'à ce qu'il ne reste plus aucune tortue-allumette. Celui qui retire la dernière allumette a gagné le jeu. Puis le programme s'arrête, on peut alors quitter le jeu.

Notons que nous pouvons quitter le jeu à tout moment en cliquant sur « cancel » dans la fenêtre de saisie qui s'affiche lorsque le joueur joue.

III) La structure de notre projet :

Nous avons décomposé le programme en trois fichiers pour rendre notre code plus modulaire, plus facile à gérer et permettre un travail de groupe plus efficace.

Deux fichiers sont des modules Python contenant les déclarations des procédures qui permettent de dessiner le décor et ses éléments. Le troisième fichier est le programme principal qui contient les instructions de notre jeu et exécute notre moteur de jeu. Les définitions des 2 premiers fichiers sont importées dans le programme principal.

1) le fichier decor_final.py

Ce fichier est composé d'une procédure principale, **dessineDecor()**, qui appelle deux autres procédures :

- **dessineVague()** dessine des demi-cercles collés les uns aux autres de façon à créer des lignes représentant des vagues.
- **dessinePoissons()** permet de dessiner les poissons et de les positionner sur l'écran d'affichage du jeu. Cette procédure appelle la procédure **dessineTriangle(longueurCote, couleur, x, y)** pour dessiner le corps et la queue des poissons.

La procédure **dessineDecor()** est appelée par le programme principal **jeudesallumettes.py**.

2) le fichier petitetortue.py

Ce fichier est composé d'une seule procédure : **dessineTortue(x)** qui dessine une seule tortue-allumette et la positionne sur l'écran d'affichage à l'abscisse x. Elle est composée de plusieurs parties : une partie qui dessine le corps et la tête de la tortue et 4 autres parties qui dessinent les 4 pattes de la tortue.

Elle est appelée par le programme principal **jeudesallumettes.py**.

3) le fichier jeudesallumettes.py

Ce fichier est notre programme principal, le moteur de jeu, qui contient les instructions et les définitions de procédures et de fonctions du jeu. Il sert à initialiser la fenêtre de jeu et affiche le décor en appelant la procédure **dessineDecor()** importée du fichier module **decor_final.py**.

Il initialise également les paramètres du jeu en dessinant les tortues-allumettes tirées aléatoirement par l'ordinateur, avec la procédure **affichageAllumettes(totalAllumettes)**. Cette procédure fait appel à la procédure **dessineTortue(x)** importé du fichier module **petitetortue.py**.

Enfin, le programme appelle la fonction **lancementJeu(totalAllumettes)** qui va nous permettre de jouer. Cette fonction fait jouer le joueur et l'ordinateur à tour de rôle jusqu'à ce qu'il n'y ait plus de tortues-allumettes. Elle retourne au programme principal le nom du gagnant : joueur ou ordinateur. Chaque fois que le joueur ou l'ordinateur joue, le nombre de tortues-allumettes restant est recalculé par appel à la fonction **tourJoueurEtOrdi(totalAllumettes, allumetteARetirer)**. Le décor est alors redessiné, et les tortues-allumettes restantes sont affichées. Les écritures du jeu affichés par la fonction **lancementJeu(totalAllumettes)** permettent au joueur de connaître, au cours du jeu, le nombre de tortues-allumettes restantes, le nombre de tortues-allumettes retiré par l'ordinateur, et permettent au joueur de saisir à son tour le nombre de tortues-allumettes à retirer.

IV) La répartition de notre travail :

L'aspect modulaire de notre programme a permis de nous répartir les tâches et le développement du code du programme.

Mathilde a créé le code d'affichage du décor et de ses éléments (la mer, les vagues et les poissons) dans le fichier `decor_final.py`.

Julien a créé le code permettant de dessiner une tortue dans le fichier `petitetortue.py`.

Quant au programme principal du jeu, en partant de la version textuelle établie lors des 2 premières séances, Julien a développé le code principal du jeu sans aucun élément de décor, afin de tester et valider la logique du moteur de jeu. Par la suite, nous avons travaillé ensemble pour assembler les différents modules que nous avons créés indépendamment, et finaliser le programme en testant et validant l'affichage graphique.

V) Les difficultés que nous avons rencontrées :

Nous n'avons pas rencontré de difficultés particulières sur la version textuelle du programme. Cependant, c'est la version d'affichage graphique qui nous a posé le plus de problèmes.

Tout d'abord, nous en avons eu dans l'affichage des tortues-allumettes car quand on lançait la procédure `affichageAllumettes(totalAllumettes)` les tortues se superposaient au lieu de s'afficher côte à côte. Pour résoudre ce problème, il a suffi de rajouter la commande `setheading(0)` dans la procédure qui a permis de réorienter le curseur de turtle sur sa position verticale originale à chaque appel de la procédure.

De plus, nous avons rencontré des problèmes lors de la première utilisation du module `decor_final.py` par le programme principal. En effet lorsqu'on lançait le jeu en version graphique, les éléments du décor comme les poissons et les vagues ne s'affichaient pas. Nous nous sommes rendu compte que cette erreur provenait du fait que la partie du code, qui devait afficher les éléments du décor, n'était pas correctement placée dans une procédure. Après modification le décor s'affichait comme attendu.

Nous avons rencontré un autre problème : lorsque le joueur commençait à jouer, lors du réaffichage du décor, les éléments du décor se décalaient vers la gauche due à une mauvaise inclinaison du curseur turtle. Nous avons donc utilisé la méthode `setheading(0)` avant chaque appel à la procédure `dessineDecor()` afin de remettre le curseur turtle dans sa position initiale à chaque tour de jeu.

VI) Comment nous nous sommes organisés ?

Nous habitons tous les deux loin l'un de l'autre. Julien rentre chez ses parents le soir et Mathilde réside à Sophia-Antipolis en résidence universitaire. La distance

nous séparant, et les séances en cours étant insuffisantes pour que l'on puisse travailler ensemble sur ce projet, nous devons avancer indépendamment sur nos programmes en dehors des heures de cours de programmation impérative.

D'où l'intérêt de décomposer le code du programme en modules afin de se répartir le développement des modules entre nous.

C'est grâce à de longues conversations téléphoniques que nous avons réussi à nous organiser et à avancer le projet ensemble, en se répartissant les tâches et en cherchant des solutions aux différents problèmes que nous avons rencontrés chacun de notre côté.

Et grâce à internet et aux e-mail, nous sommes parvenu à nous tenir tous les deux à jour par rapport au travail de l'autre en partageant le projet après chaque tâche accomplie ou chaque modification du programme.

VII) Quel est le travail qu'il reste à faire ?

Toutes les consignes de travail du professeur sont remplies et l'objectif pédagogique du projet est atteint.

Cependant, nous pourrions encore améliorer notre programme pour le rendre plus attractif. Nous pourrions améliorer le « design » des animaux aquatiques, en ajoutant des éléments comme des écailles, des yeux et une bouche aux poissons, dessiner une belle carapace colorée aux tortues. Tous ces petits détails que nous aurions pu faire si nous avions eu plus de temps.

Et enfin réaliser la version + et ++++.