Rapport Projet Programmation

Vous retrouverez ici notre rapport sur le projet de programmation, c'est à dire les règles, buts et intentions tout d'abord, puis comment nous avons structuré celui-ci (fonctions, etc...), la répartition du travail entre nous, les difficultés que nous avons rencontré ainsi que les solutions que nous avons mis en place pour les régler.

I/ A propos du jeu

Notre Jeu s'appelle "le jeu du château". A raison, puisque le but principal est de battre notre adversaire d'un royaume voisin afin de prendre possession du château, un point stratégique. Ainsi, il faut être celui qui détruit la dernière tour, pour en première ligne pour courir et s'emparer du château.

Le jeu se déroule au tour par tour, d'abord nous, puis notre adversaire. On choisit la ligne (ou groupe consécutifs) de tours, puis la position ou l'on veut jouer (gauche, milieu, droite). Lorsque l'on joue sur les côtés, une tour est détruite. Lorsqu'on joue au milieu, deux tours sont détruites. Le jeu est donc un jeu de stratégie.

Le décors est composée de plusieurs éléments : Un sol mimant une plaine, des sapins, un ciel avec des nuages, des rayons de soleil traversant le ciel. Il est aussi composé de plusieurs constructions : Un grand château en plein milieu, qui est l'objectif du jeu, et des tours, qui se casseront au fil de la partie.

Notre jeu a été programmer sous le langage de programmation interprété, multi-paradigme et multiplateformes Python.

II/ Structure du jeu

Notre Jeu est structurées en plusieurs modules contenant toutes les fonctions et lignes de codes permettant de jouer. Ils sont au nombres de 8 et ont tous des utilités différentes.

- Le moteur du Jeu :

Ce modules contient toutes les fonctions permettant au joueur de jouer (joueurJoue), a l'ordinateur de jouer (joueurJoue), donc d'afficher les Tours sur le terrain, (afficheQuille), selon si le joueur ou l'ordinateur a joué au milieu ou sur les côtés (jouerMilieu et jouerCôté). Ce module est l'un des plus importants car il est celui permettant à tout le jeu de fonctionner correctement.

- Les modules "Dessin":

Il sont au nombres de 6 et permettent de dessiner le décors et les tours. Ils sont un éléments essentiels afin d'avoir un jeu certes jouable, mais aussi appréciable visuellement.

-Les Polygones et déplacements :

Ce module comprend toutes les formes de base qui sont utilisé dans tous les autres modules "Dessin" ainsi que des déplacements simplifiés fait par la turtle.

- Pour ce qui est des formes de bases, il est composés de fonctions permettant de dessiner des rectangles, des polygones réguliers, des parallélépipèdes et des trapèzes (respectivement les fonctions dessineRectangle, dessinePolygone, dessineParallélépipède, trapeze).

- Pour ce qui est des déplacements, il a une fonction qui lui permet d'aller au sommet gauche d'un trapèze et d'une fonction qui lui permet d'aller au coordonnées indiquées sans faire de traces sur le dessin (respectivement hautTrapeze et aller.)

-Les éléments du château :

Ce module comprend tous les éléments autres que les murs, etc... qui composes le gros chateau ainsi que les tours.

Ce module utilise donc des fonctions du module "polygones et déplacements".

Il contient donc les fonctions permettant de faire :

- Les briques, les briques tombées et les drapeaux blanc pour les tours (respectivement les fonctions briqueTour, briqueTombée, drapeauxBlanc)

- les drapeaux et les créneaux pour les deux (respectivement les fonctions drapeauxTourChateaux et créneaux)

- Les bannières, les portes et toit pour le château principale (respectivement les fonctions bannièreChateau, porte et maison, et toit)

-Les Polygones avec ombrages :

Ce module comprend tous les polygones nécessaires pour donner du relief aux éléments du décor, c'est a dire des polygones avec un dégradé du plus clair vers le plus foncé, de gauche a droite.

Ce module utilise donc des éléments des modules "polygones et mouvements" et "éléments du château".

- Pour ce qui est des tours et du chateaux, il est composés des fonctions permettant de mettre du relief sur les trapèze, les rectangles, les crénaux (dégradé de gris, respectivement les fonctions trapezeOmbreBase, rectangleOmbreBase, rectangleOmbreSousToit, toitOmbre ), les toits (dégradé de rouge, respectivement la fonction toitOmbre)

- Pour ce qui est des sapins, il est composés des fonctions permettant de mettre du relief sur la partie "feuillage" et sur le tronc des sapins (respectivement les fonctions sapinOmbre et troncOmbre)

- Les Tours :

Ce module comprend les deux fonctions qui permettent de construire de construire les Tours debout et détruites, ainsi que la fonction qui permet de les placer sur le terrain (utilisé par le module "moteur du jeu").

Ainsi ce module utilise des fonctions des modules "polygone et mouvement", "éléments du château" et "polygone ombre".

- Les tours debout sont constituées des fonctions trapeze, trapezeOmbre, créneauxOmbre, briqueTour, et drapeauxTourChâteaux, et se déplace grâce aux fonctions aller et hautTrapeze.

-Les tours cassée sont constituée des fonctions dessineRectangle, trapezeOmbre, créneauxOmbre, drapeauxBlanc, briqueTourCassée

- La fonction de placement des tours est composée de trois boucle while, une commune et deux qui dépendent du nombre, pair ou impaire, de tours sur le terrain est pair ou impaire. Cela a pour but de pouvoir les placées en quinconces, pour que ça soit plus joli.

Toutes les tours ont été codées de telle sorte a pouvoir être placé n'importe ou et selon n'importe quel gradissement.

-Le gros château :

Ce module est utilisé pour faire le gros Châteaux, qui constitue le prix de notre jeu. Il contient donc des fonctions permettant de faire chaque tour particulière du château, la porte du château ainsi qu'une fonction les regroupant toutes permettant de faire le château en entier.

- Les tours en elles mêmes utilise a peut près toute une fonction nommée baseTour(sauf celle du milieu et la grosse a gauche), permettant de simplifier l'écriture des tours. Elles sont sinon constituées des fonctions rectangleOmbre, trapezeOmbre, rectangleOmbreSousToit, Toit, crénaux, drapeauxToursChateau et baniereChateau. Celles-ci on été faite de telle sorte que l'on puisse les mettre ou l'on veut et les agrandir ou rétrécir selon les besoin.

- La porte d'entrée est on ne peut plus simple. Elle est simplement composée des fonctions porte, maison et baniereChateau.

- La fonction gros château place toutes les tours entre elles et est faite de telle sorte que l'on puisse choisir où l'on veut le château et le grandissement qu'on lui veut.

- Le fond :

Ce module contient toutes le fonctions permettant de faire le décor de notre jeu. Donc le ciel, les rayons de soleil, les nuages, le gros château, le sol et les sapins.

Ce module a besoin des modules "polygones et déplacements", "polygones ombres" et "gros château".

- Le ciel est composé du background mis a la couleurs bleu, de rayons de soleil fait a base de trapeze, de nuage (utilisant une fonction pour faire les nuages et une autre permettant de faire de l'ombre dessus).

- Le sol est d'abord constitué du gros château, puis du sol (plusieurs polygones réguliers de plusieurs couleurs différentes avec énormément de côtés) puis des sapins (composées des fonction sapinOmbre, troncOmbre et placementSapin).

Ce module contient aussi une fonction "cache" permettant de cacher les tours et le texte entre chaque tours, afin de pouvoir évoluer dans la partie. Il est composé du sol, des sapins et d'un nuage.

Le module "exécuteur" du jeu :

Il est le module permettant de lancer le jeu. Il est celui qui place le texte sur le terrain pour le nom, les règles du jeu, Le tour de quel joueur ainsi que le nombre de ligne(s) (groupe(s)) de tour qu'il y a sur le terrain.

Il est le module qui importe tous les autres modules vu précédemment (même si certains ne sont pas utilisés, c'est un peu pour avoir un mémo de tous les modules que contient le jeu)

Il permet à chaque joueurs de jouer chacun l'un après l'autre, en demandant le numéro de la ligne (du groupe) de tour que l'on veut puis de la position(gauche, droite ou milieu) de la tour que l'on veut détruire.

Avant chaque tour il applique le cache vu dans le module précédent afin de masque le tour précédents. Il affiche ainsi a chaque tour le nombre de groupe de tour qu'il y a.

Il est aussi le module qui permet de décider, entre 15 et 20, le nombre de tour qu'il peut y avoir dans le jeu.

A la fin de chaque partie, il affiche un message de victoire ou de défaite, suivant qui est celui qui a détruit la dernière tour.

III/ Déroulement et répartition du travail :

Notre Jeu a pris forme au travers de plusieurs étapes, c'est a dire en commençant par la version textuelle, puis par le choix de la forme des quilles et fonctions nécessaire pour la version graphique, ensuite par la modification de la version textuelle pour placer le jeu dans turtle, et enfin par la création de tout le décors et de la mise en place de l'histoire.

Version textuelle:

La version textuelle fut la première et l'une des étapes les plus dur de notre programme, car elle est celle sur laquelle repose l'entièreté du jeu. Nous avons essayé au maximum de le faire ensemble, soit via message et mail interposés, soit en se retrouvant à la bibliothèque afin que chacun de nous sache comment le jeu fonctionne, pour être sûr de ne pas faire n'importe quoi par la suite. Nous avons donc globalement (même si certaine choses ce sont fait en travail personnel) fait les choses tous les deux.

Fonction de base et choix du thème :

Le thème est une idée de Benoît validée par Lillian qui trouvait l'idée sympathique. Cependant nous avons mal commencé puisque nous avons essayer de dessiner les quilles sans fonction au départ, ce qui, on l'a vite compris, fut une erreur stratégique notable. Nous nous sommes donc posés tous les deux pour savoir quelles seraient les fonctions dont on auraient besoin, ne serait-ce que pour constituer simplement les éléments du décors (polygones, etc...).

Modification de la version textuelle :

La première étape pour la modifier fut de constitué une première tour "de base" droite et une première tour "de base" détruite a modifier plus tard simplement pour permettre de convertir les quilles de textuel en turtle. Lillian s'occupa de constituer les fonctions de base (trapeze, rectangle)

pendant que Benoît s'est occupé des créneaux. Puis Lillian constitua la première version de la tour droite en suivant les plan prévu pendant que Benoît eu pour tâche de designer la tour casser puis de la réaliser. L'idée fut d'abord validé en version papier par Lillian puis il put la faire par python.

Ensuite, Lillian créa la première version du placement des quilles (version linéaire), puis Benoît s'occupa de l'intégrer dans la version textuelle de jeu (c'est a dire modifier la version textuelle et faire en sorte que les tours s'affiche bien correctement a chaque tour, puis que le progrmamme demande dans turtle ce que le joueurs veut jouer).

Création de tout le décors et de la mise en place de l'histoire :

Une fois que le jeu fonctionna dans turtle, nous avons pu nous attaquer au décors ainsi qu'a faire les version final des tour debout et cassée du placement des quilles (pour qu'elles soient en quinconces).

Comme Lillian avait un peu de mal avec les couleurs, il s'est occuper de faire une première version sans nuance des sapin, des nuages et du sol , pendant que Benoît c'est chargé de faire les polygones avec des nuance pour pouvoir rajouter de la nuance aux éléments de décors de Lillian.

Comme les polygones avec ombres ne furent pas très longs a faire, Benoit s'occuper de faire le château.

Lillian s'occupa des rayons de soleil pendant que je l'occupait de modifier les tours de bases en remplaçant les polygones de base par des ombrés, en ajoutant un peu de texture (brique) sur la tour droite, et enfin en ajoutant des drapeaux (drapeaux de base fait par Lillian puis modifiés par Benoît).

Globalement, Lillian s'occupa des formes de bases et des première version du placement des éléments (sauf pour le château) tandis que Benoît c'est occupé de la partie couleur et nuances, et versions finale des placements.(sauf pour le château ou il fait les formes de bases puis les nuances de couleurs, car ils n'ont pas pu échanger pendant les vacances).

On a donc finit par ajouter tous les éléments ensemble dans le programme de base, puis nous avons ajouté le texte et enfin nous avons tout séparés en modules différents.

IV/ Problèmes rencontrés et solutions apportées

Version textuelle:

Nous avons surtout rencontré des difficultés dans la version textuels. Certains éléments comme l'affichage de la dernière quille ou alors le fait de jouer au milieu ne voulaient pas fonctionner correctement (bug car fonctionne d'une façon puis change de façon de fonctionner alors que nous n'avons toucher a rien). Alors après s'être acharner dessus, nous avons décider de recommencer un fichier de la version textuels, pour se rendre compte qu'en fait le fichier de base avait été corrompu. Ainsi la solution a été de recommencer, et de tester chaque étape une par une a l'aide de print pour vérifier que tout le processus marchait correctement.

Fonction de base et choix du thème :

Difficultés rencontré sur une des fonction de bases : le trapèze. Nous avions commencé ensemble

mais elle ne marchait pas, alors Lillian c'est occupé de la faire chez lui. Au final il a trouvé que nous avions des problèmes au niveau des angles et que nous avions oubliés des valeurs abolues. Il y a donc remédié en ajoutant tout ce qui manquait.

Modification de la version textuelle :

Le majeur problème fut rencontré lors de la mise en commun du placement des quille et de la version textuelle. En essayent de rajouter le placement des quille dans la fonction afficheQuille, cela ce fonctionnait pas, les quilles se plaçait toujours a des endroits pas prévu, souvent hors de l'écran. Pour y remédier, Benoît a décider de laisser les deux séparer, et simplement de rediriger afficheQuille vers le placement des tours. Cela en plus a lui a permis de pourvoir modifier le placement des tours plus facilement.

Création de tout le décors et de la mise en place de l'histoire :

Nous n'avons rencontré que peu de problème sur cette étapes. Le seul fut avec les toit en relief, qui durent être modifiés car ne donnait pas un bon résultat.

Une difficultés apparue pour la mise en place des texte dans le jeu. Le problème étant que la fonction write n'est pas comme la fonction print, ne permet pas d'afficher correctement le texte ("il y a " len (q) "groupes"). La solution a donc été de mettre plusieurs fonctions write pour un seul et même texte.

Le seul problème non résolu de notre programme est le fait qu'il est trop complexe pour qu'il soit rapide. La fonction tracer (0,0) n'est même pas suffisante (permet quand même de beaucoup accélérer), mais le jeu ralenti au fur et a mesure a cause des sapin notamment, qui sont assez complexes au niveau des couleurs.

Sinon, pour le reste, le jeu fonctionne très bien.

V/Conclusion

Ce fut une expérience très agréable pour tous les deux, une très bonne introduction a la programmation de choses complexes. Nous espérons surtout que notre programme vous plaira autant qu'a nous.

Merci beaucoup de nous avoir lu !