

Ariane & le Minotaure

Rapport du projet

Table des matières :

- Objectifs du projet
- Organisation du programme
- Fonctions
- Difficultés rencontrées
- Organisation / répartition du travail

Objectifs du projet

Pour aboutir au projet du fameux «Ariane et le Minotaure », nous avons pour objectifs :

- de transformer un fichier texte en une liste de liste;
- faire des niveaux prédéfinis et aléatoire ;
- savoir extraire des fichiers qui sont dans un répertoire ;
- permettre à l'utilisateur de remettre le labyrinthe à zéro ;
- définir les coups interdits ;
- détecter lorsque le joueur gagne ou perd;

Organisation du programme

Notre programme est organisé en deux parties : un fichier contenant les différentes maps et le code qui utilise c'est maps pour jouer au jeu. C'est-à-dire que lorsque l'utilisateur va lancer le jeu, le code va venir prendre la map correspondante au niveau sur lequel le joueur est. La map est composée de personnages qui se déplacent et de mur. Pour réussir un niveau, le joueur doit réussir à ramener Thésée et Ariane sur la porte sans que les minotaures ne bloquent le chemin mais aussi les mur. Cependant les mur ne peuvent pas être déplacés par le joueur. Si le joueur ne veut pas se faire attraper, il faut qu'il soit séparé d'un murs pour ne pas être atteint sinon le minotaure vertical est sur la même ligne que lui sans obstacle celui -ci l'attrapera lors de son tours. C'est le même raisonnement pour le minotaure horizontal : s'il se trouvent sur la même colonne qu'Ariane et sans obstacle face à lui, il pourra vous attraper si c'est à son tour de jouer.

Le programme doit aussi empêcher les coups illégaux : Ariane ne peut pas traverser les murs ni les minotaures; et doit afficher un message lorsque l'utilisateur a gagné le puzzle.

Mais il peut aussi choisir de jouer avec le mode aléatoire : le plateau est affiché de manière aléatoire et grâce au solveur (qui permet de vérifié si un plateau aléatoire est correcte) le joueur peut jouer à ce niveau en étant sûr d'avoir un chemin correct.

Toutes les valeurs initiales du programme sont mises dans une même fonction pour bien rendre le code lisible sans avoir à trier entre valeur et fonction.

Fonctions

Pour lancer le programme, nous l'avons incrémenté de plusieurs fonctions. Ces fonctions sont plus ou moins importantes pour le lancement du jeu.

Voici quelques unes des fonctions principales du jeu :

```
def est_permis(plateau, perso, direction) :  
    Dans cette fonction on calcule la nouvelle case vers  
    laquelle Ariane va se déplacer. On vérifié si elle ne  
    dépasse pas la taille du plateau. On récupère la case du  
    mur, c'est à dire la case qui se trouve entre la case  
    d'Ariane et celle où elle se déplace. Si la nouvelle case  
    est un minotaure, ou s'il y a un mur entre les 2 cases,  
    on renvoie Faux sinon Vrai.
```

```
def deplacer_perso(plateau, perso, direction):  
    Dans cette fonction on calcul les nouvelles coordonnées  
    après une certaine direction donnée en paramètre, la  
    nouvelle case correspond à la lettre du personnage et  
    on remplace "l'ancienne" position du personnage par un  
    espace. Pour finir on met à jour les coordonnées du  
    personnage.
```

Difficultés rencontrées

En réalisant se projet, nous avons rencontré plusieurs difficultés :

- Tout d'abord la difficulté principale était le solveur. En effet la manipulation des matrices pour le solveur et les différentes conditions à suivre nous ont pris pas mal de temps et nous n'avons malheureusement pas pu fournir un solveur.
- De plus, les déplacements des minotaures étaient lié à ceux d'Ariane donc il fallait faire attention à comment on modifiait les coordonnées et leur direction en fonction de leur position et celle d'Ariane.
- Et l'autre difficulté que nous avons rencontrée était d'afficher deux personnages sur une même case, car au départ on affichait leurs photo lorsque leurs lettre apparaissait dans le plateau. On ne pouvait donc pas mettre deux lettres dans une case. On a donc décidé d'afficher nos personnage en fonction de leur coordonnées qui sont actualisées à chaque déplacement.

Organisation / répartition du travail

Pour la répartition du travail, nous avons décidé de travailler ensemble sur chaque point du projet pour pouvoir avancer plus vite sur le travail à fournir.

De plus, nous mettons nos programmes à exécution pour aider à corriger un bug et/ou une erreur non vu par celle qui a écrit le programme.