

# Rapport d'avancement SAé 2.1

## Introduction

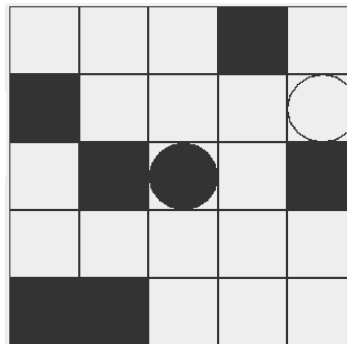
Dans le cadre de cette SAé, il nous a été demandé de créer deux algorithmes de résolution de labyrinthe : l'un aléatoire, l'autre déterministe. Le point de départ est considéré comme Thésée. A cela s'ajoutent plusieurs fonctionnalités telles que la génération ou le choix du labyrinthe, sa sauvegarde ou le mode de déroulement de l'algorithme.

## Fonctionnalités

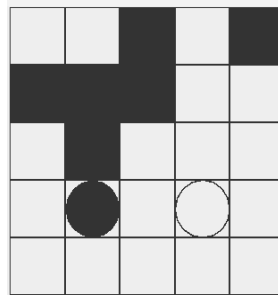
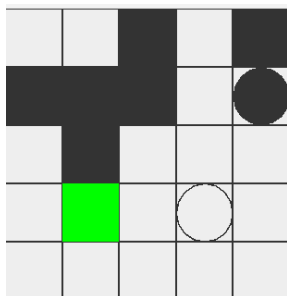
Notre programme ouvre tout d'abord un menu demandant de créer une grille, remplie aléatoirement ou vide, ou bien de sélectionner un fichier contenant les informations d'une grille existante.

Une fois la grille créée, la fenêtre nous offre une représentation graphique de la grille :

- Thésée est un cercle noir
- la sortie est un cercle blanc
- un mur est un carré noir
- une case libre est un carré blanc



Elle pourra être entièrement modifiée. Pour cela, il faudra cliquer sur une case. Elle changera alors de couleur, à part si Thésée ou la sortie sont dessus. Dans ce cas-là, le déplacement de Thésée ou de la sortie sera en attente, et un clic sur une autre case les déposera sur celle-ci.



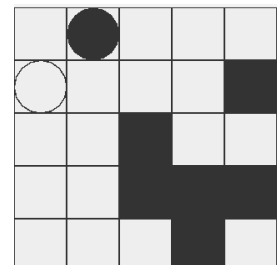
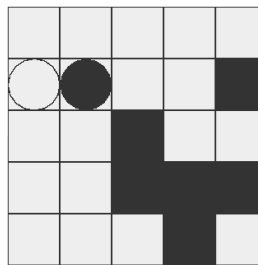
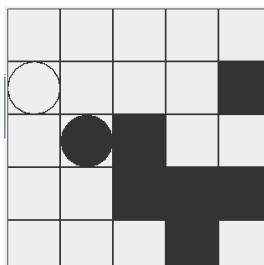
à la gauche de l'écran se trouvent différents boutons. On pourra recréer, comme sur le menu principal, une nouvelle grille.

En plus de cela, on pourra également appliquer à la grille actuelle l'algorithme aléatoire ou l'algorithme déterministe.

Dans le premier cas, 100 instances de l'algorithme aléatoire seront lancées. La moyenne des tentatives sera calculée puis affichée dans le menu.

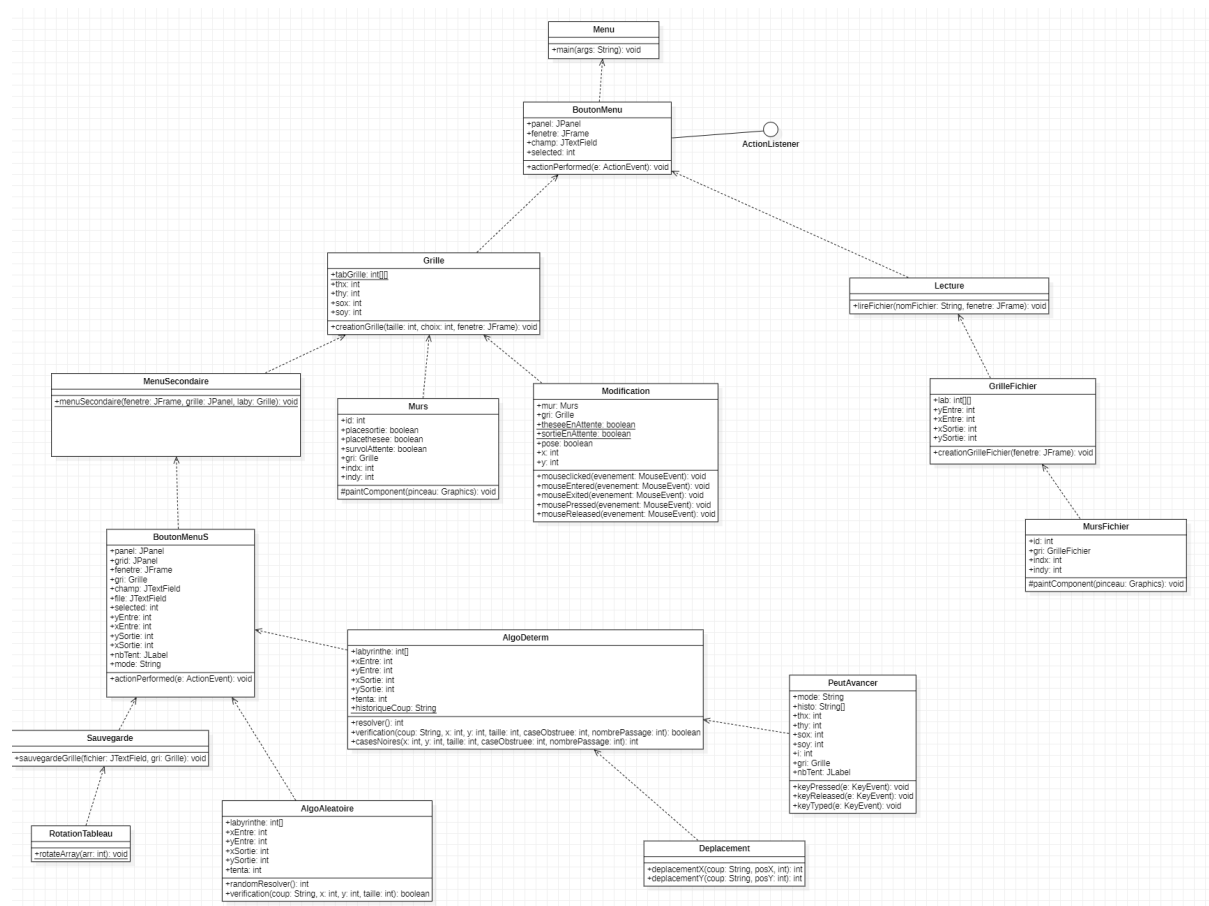
Dans le second cas, il sera possible de choisir le mode de défilement de l'algorithme déterministe. Si il est automatique, le nombre de tentatives est directement affiché.

Dans le cas où le mode de défilement est manuel, l'utilisateur pourra visualiser le déplacement de Thésée en appuyant sur une touche du clavier. Une fois l'algorithme terminé, Thésée reviendra au point de départ.



Enfin, l'utilisateur pourra sauvegarder la grille en l'état actuel. Il pourra ainsi choisir un nom de fichier.

## Diagramme de classe



## Algorithme déterministe

L'algorithme que nous avons choisi est basé sur l'algorithme en profondeur utilisé dans les graphes.

Nous avons donc dû choisir ses priorités de déplacements, qui sont dans l'ordre :

- droite
- haut
- gauche
- bas

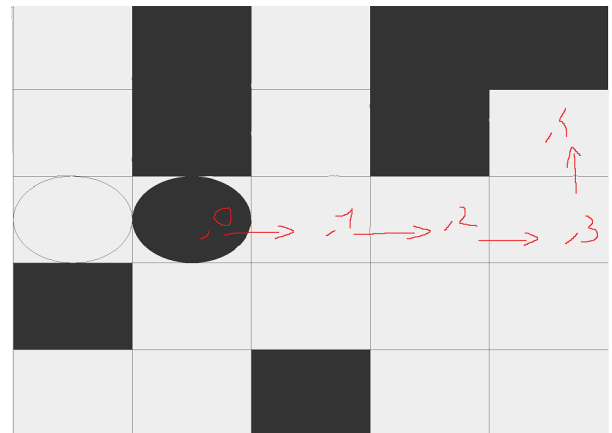
Voici en photo les étapes du déroulement de l'algorithme.

### Étape 1 :

On voit qu'il part vers la droite en ignorant les autres directions.

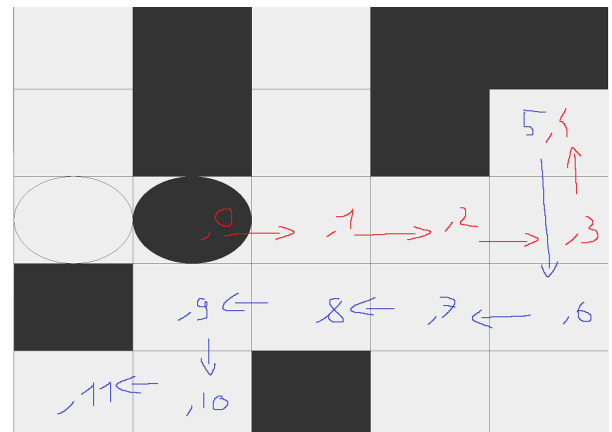
Lorsque la droite n'est plus disponible, il choisit le haut car celui-ci a la priorité sur le bas.

Ici, il est bloqué, il doit rebrousser chemin en désempilant la case qui ne mène à rien.



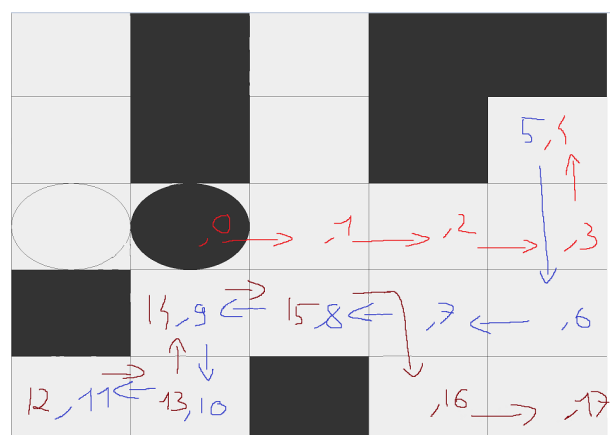
### Étape 2 :

En bleu le chemin parcouru lors de la deuxième étape. La case ,3 n'est pas désempilée car il pouvait encore explorer une case en bas.

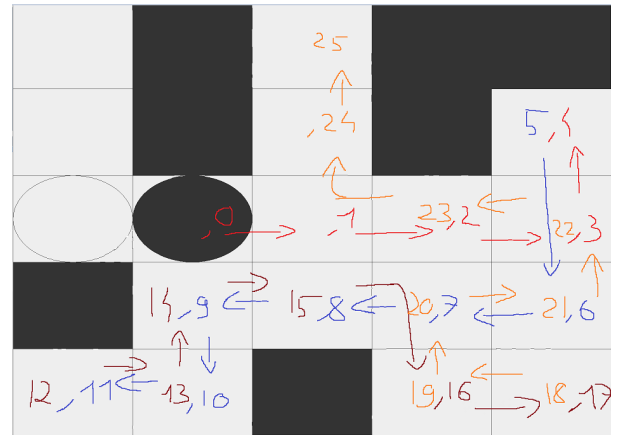


### Étape 3 :

De la même façon, la case ,7 n'est pas dépilée car elle permet d'explorer une nouvelle case.



Rien de particulier, l'algorithme suit la même logique



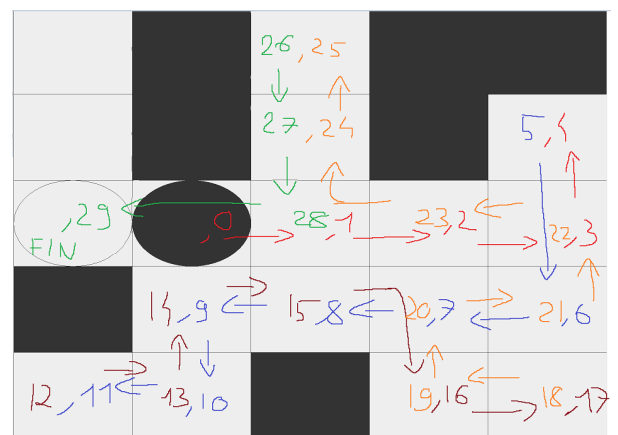
### Étape 5 :

Ici, l'algorithme trouve la sortie, donc il s'arrête et il aura trouvé au bout de 30 coups.

Et ce sera **toujours** 30 coups pour les autres tentatives.

Si l'**algorithme** dépile la case de départ, le labyrinthe sera considéré comme **impossible**.

En effet, avec cette logique, **l'intégralité** des cases atteignables seront atteintes dans le pire des cas.



## Conclusion

Oscar : Ce projet a été pour moi une bonne expérience car j'avais plus de mal en java qu'en C, mais il m'a permis de m'améliorer grandement et de comprendre les notions principales de la programmation objet même si j'aurais voulu plus perfectionner le programme, faute de temps.

Edouard: J'ai trouvé que ce projet était plus intéressant que le projet C, notamment de par l'utilisation des fenêtres. J'ai pu grandement m'améliorer sur l'utilisation des ActionListener ou des lectures de fichiers, même si ce dernier m'a donné du fil à retordre.