

Rapport Projet 2.1

L'algorithme d'Arianne

Élèves responsables du projet :

Louis Chaillou

Tanguy Gabriel

I) Introduction

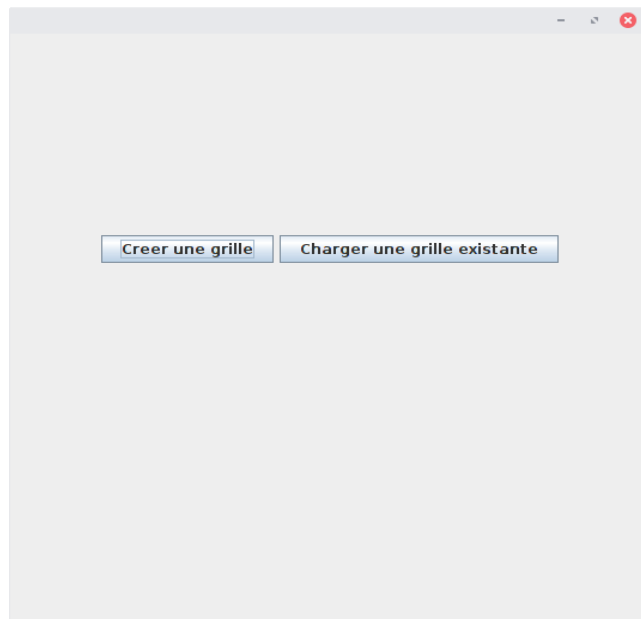
Le projet a pour but d'étudier l'évolution d'un algorithme de guidage sur une grille préalablement déterminé, à conduire un objet mobile (dans ce projet un carré) jusqu'à son but à travers un parcours d'obstacles (sortie symbolisée elle aussi par un carré). Le carré devra donc trouver la sortie en utilisant un type d'algorithme.

L'utilisateur pourra choisir de déterminer quelle grille il choisit pour pouvoir avancer et par la suite la personnaliser comme il le souhaite (position des obstacles, du carré responsable de l'algorithme appelé Thésée, de la sortie, il pourra s'il le souhaite mettre de positionner de manière aléatoire des cases noires et puis par la suite de sauvegarder sa grille.

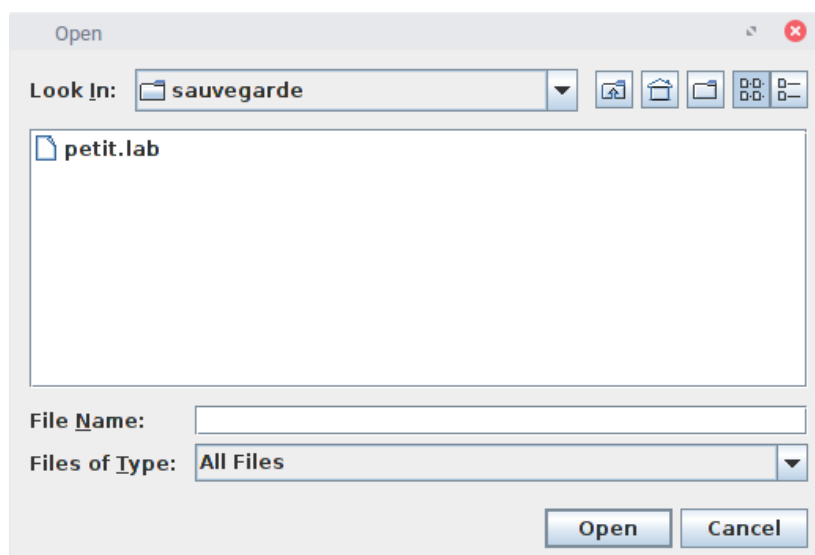
Par la suite, il déterminera si l'algorithme pourra s'effectuer de manière aléatoire (le programme renverra seulement le nombre de déplacement du carré pour trouver la sortie) ou de le faire de façon manuelle (étape par étape avec une visualisation des déplacements).

II) Description des fonctionnalités

- Pour choisir la grille, l'utilisateur pourra construire une nouvelle grille.



- Ou charger une grille existante



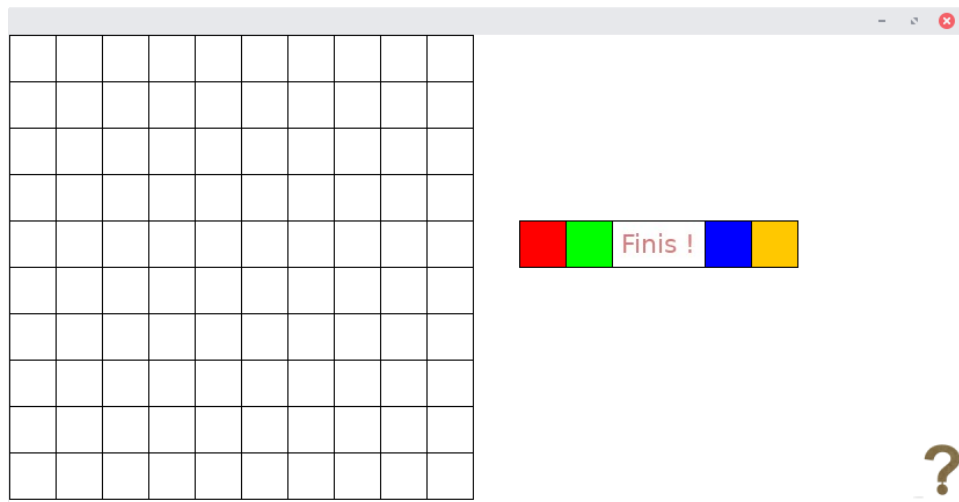
- S'il choisit la première option, l'utilisateur pourra décider d'abord de la taille de la grille. L'utilisateur pourra ensuite choisir entre deux algorithmes : l'un, totalement aléatoire (chaque direction a autant de chances d'être choisie), et l'autre déterministe.



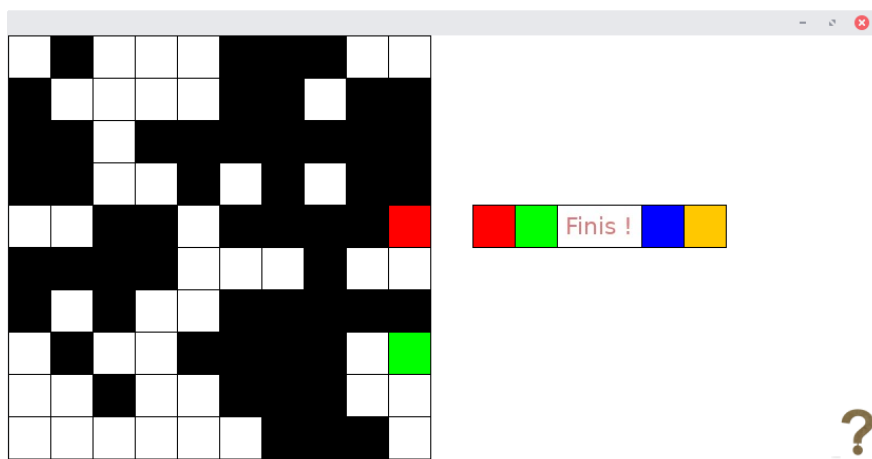
Entrez la taille de grille souhaité puis choisissez un algorithme

A.Aleatoire A.Deterministe

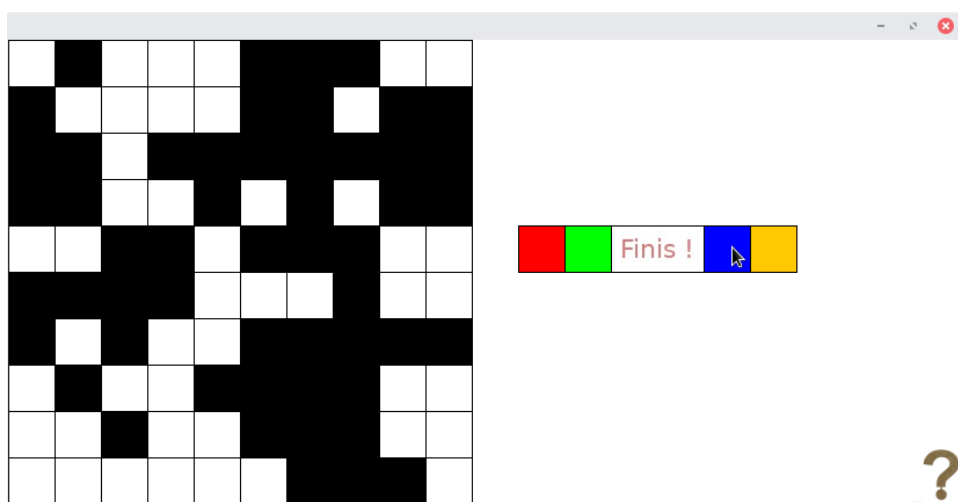
- Il pourra ensuite partir d'une grille vide



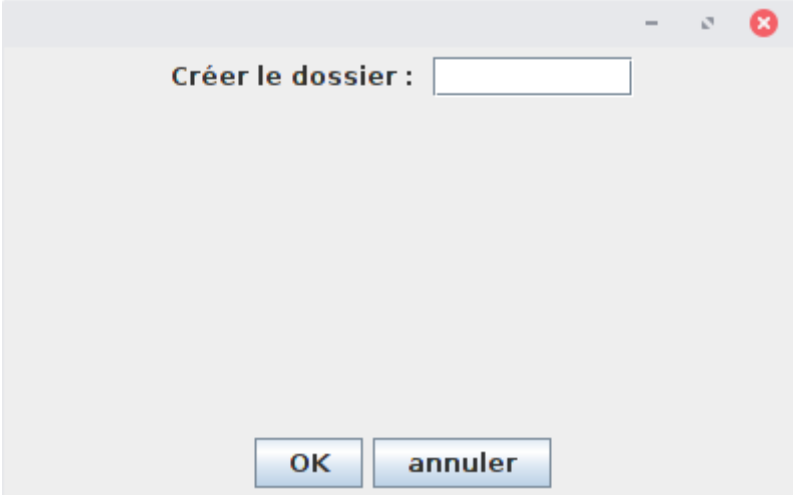
- Posez les obstacles, Thésée et la sortie



- Il peut disposer des obstacles de façon aléatoire en appuyant sur la case bleue



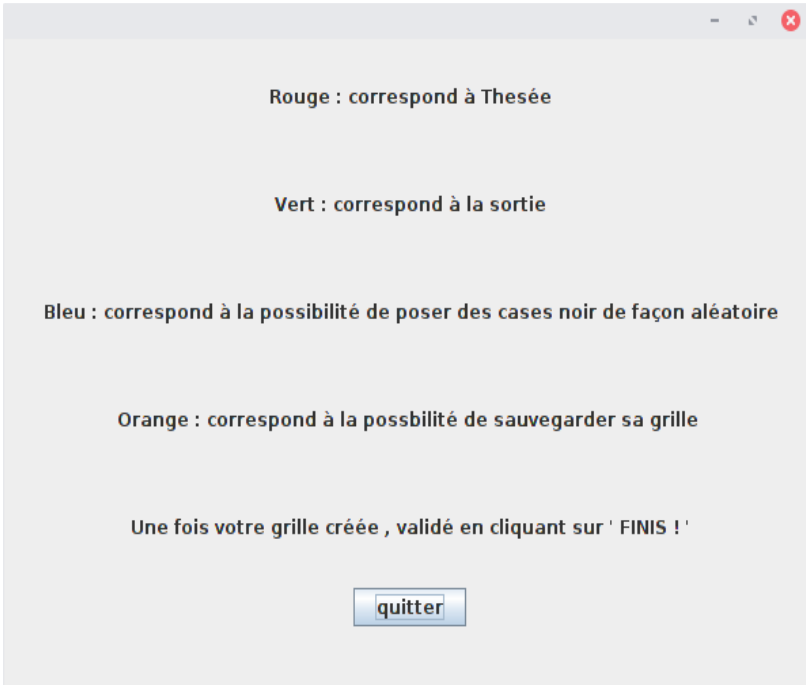
- Une fois la grille terminée, l'utilisateur aura la possibilité de la sauvegarder en appuyant sur le carré orange (sous un nom de son choix au format .lab)



Créer le dossier :

OK annuler

- L'utilisateur peut consulter la légende s'il souhaite (en appuyant sur le point d'interrogation)



Rouge : correspond à Thesée

Vert : correspond à la sortie

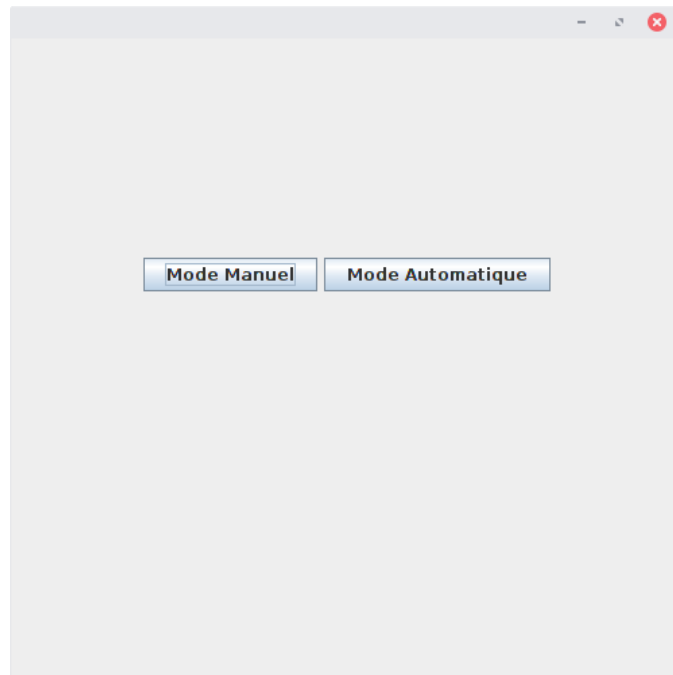
Bleu : correspond à la possibilité de poser des cases noir de façon aléatoire

Orange : correspond à la possibilité de sauvegarder sa grille

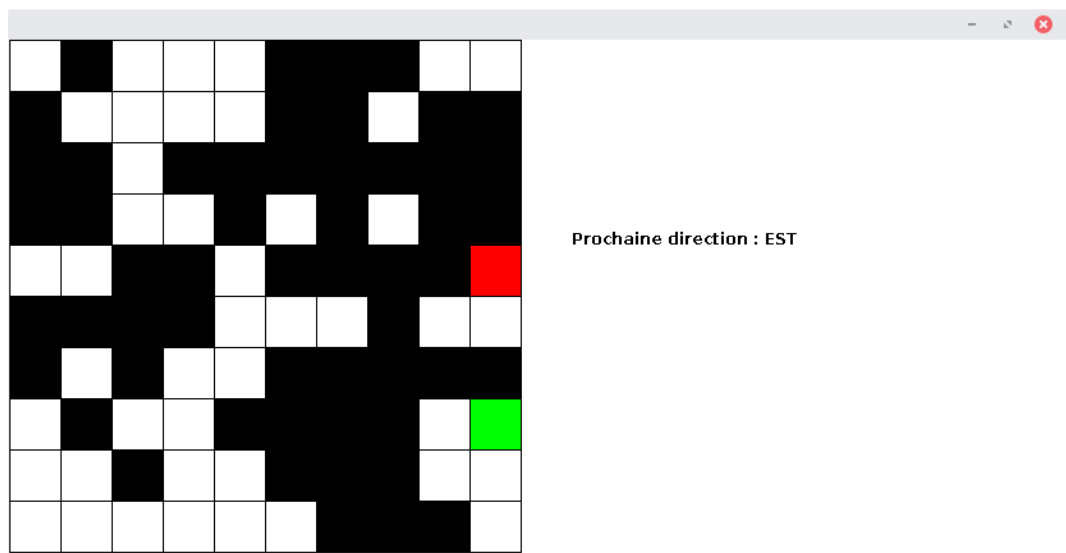
Une fois votre grille créée , validé en cliquant sur ' FINIS ! '

quitter

- Une fois l'algorithme et la grille choisie, l'utilisateur devra sélectionner une visualisation du déroulement de la simulation : manuelle ou automatique.



- Dans le mode manuel, l'avancement de Thésée sera représenté sur la grille, et le choix préconisé par l'algorithme pour la prochaine étape sera affiché. L'utilisateur fera défiler les étapes en appuyant sur une touche, jusqu'à la fin de la simulation. Ce mode sert à voir en détail le comportement de l'algorithme.



- Dans le mode automatique, la grille n'est pas affichée, et la simulation est effectuée sans intervention de l'utilisateur. On affichera seulement le nombre d'étapes nécessaires pour compléter la simulation. Dans le cas de l'algorithme aléatoire, on fera tourner la simulation 100 fois avant d'afficher le nombre d'étapes moyen (puisque le résultat sera différent à chaque simulation)



III) Présentation de la structure du programme

Voici un diagramme représentant la structure du programme :

Voir dans le git fichier : Main.svg

IV) exposition de l'algorithme déterministe

L'algorithme déterministe a été créé pour déterminer la sortie de façon intelligente gardant en mémoire ses déplacements. Il teste alors les quatre directions à la suite : le nord, l'est, le sud et l'ouest.

S'il peut accéder à la case qu'il souhaite en fonction de la direction alors il enregistre dans un tableau qu'il est déjà passé par cette case, dans un autre quelle position il a pris à cette case, il enregistre aussi en fonction de nord, est, sud, ouest qu'il n'a pas le droit de retourner sur la position de son père. S'il ne peut pas accéder à la position car il y a une case noire alors il marque dans un tableau de reconstruction la position à partir de laquelle cette case noire se trouve pour s'en souvenir et ne pas faire appel à un autre programme, il passe ensuite à la direction d'après.

S'il est bloqué entre case noires ou s'il doit prendre des positions qu'il a déjà pris alors il utilise son tableau qui fonctionne en fonction du compteur là où il a la position précédente pour y retourner. Il ne reteste pas toutes les directions de cette case précédente car il a déjà connaissance de la direction qu'elle avait prise et donc elle recommence à partir de là. Bien sûr cet algo n'a pas connaissance du tableau de case noires il se sert de l'analyse autour des cases pour mettre en œuvre sa reconstruction.

v)Conclusion

Par ma part, la complexité du projet m'a permis d'en apprendre davantage sur mes compétences et mon réel savoir-faire.

De nombreuses problèmes ont permis de mettre ma patience et ma réflexion à rude épreuve.

Ce fût un plaisir de collaborer avec mon camarade et de terminer le projet.

Louis chaillou : J'ai beaucoup aimé travailler sur ce projet car il était très différent du premier, surtout parce que ce n'était pas un jeu. J'ai trouvé qu'il était beaucoup plus technique mais cela m'a permis de beaucoup m'améliorer en java et d'acquérir une logique très enrichissante pour aujourd'hui comme pour demain.