

Projet CY-PATH

v1.0

CLASSE ING1-GI • 2022-2023

AUTEUR Romuald GRIGNON

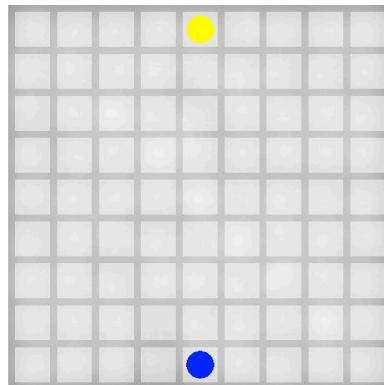
E-MAIL romuald.grignon@cyu.fr

DESCRIPTION

Ce projet est un jeu de plateau compétitif pour 2 ou 4 joueurs. Le but est de faire traverser le plateau à son pion avant les autres joueurs.

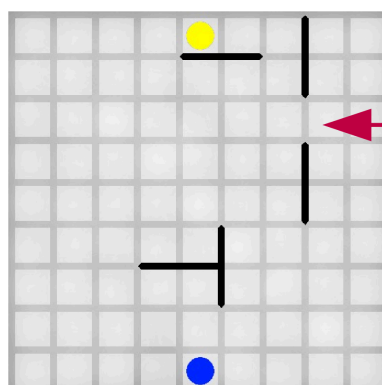
La zone de jeu est constituée de 81 cases disposées en carré. Chaque case est séparée des autres par des lignes horizontales et verticales (quadrillage).

Chaque joueur démarre sur la case milieu d'un des bords du plateau (pour 2 joueurs, les pions doivent être placés sur des bords opposés).



Chaque joueur joue chacun son tour. A chaque tour de jeu, un joueur peut choisir de déplacer son pion d'une seule case horizontalement ou verticalement, dans n'importe quelle direction.

Il peut également choisir de positionner une barrière, horizontalement ou verticalement. Une barrière possède une longueur de 2 cases. Une barrière ne peut pas dépasser de la zone de jeu. Une barrière ne peut pas superposer une autre barrière. Une barrière est forcément placée pour avoir 2 cases adjacentes complètes. Les barrières sont au nombre de 20 au total.



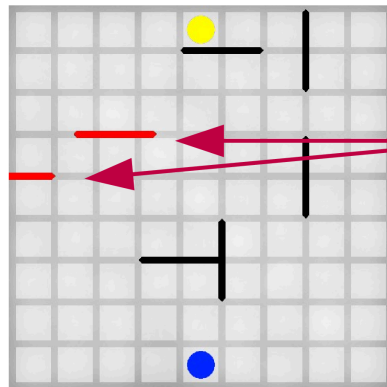
Impossible de mettre une nouvelle barrière verticale ici, car elle se superpose à d'autres barrières

Les barrières ne peuvent pas être franchies par les pions des joueurs. Ces derniers devront donc trouver un autre chemin pour progresser en direction du bord opposé.

Lorsqu'une barrière est posée, il est impossible de la déplacer pour le reste de la partie. Lorsqu'il n'y a plus de barrières à poser, les joueurs ne peuvent alors que déplacer leurs pions pour terminer la partie.

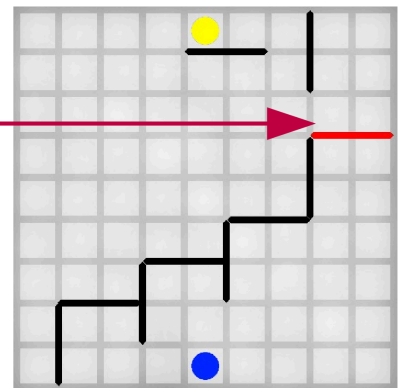
Si il n'est plus possible de poser de nouvelles barrières, alors la seule action possible pour un joueur est de déplacer son pion.

Lorsqu'un joueur pose une barrière, il doit s'assurer que le pion du joueur adverse dispose au moins d'un trajet possible. Il est interdit de bloquer définitivement l'adversaire : tout joueur doit avoir au moins 1 trajet possible.



Impossible de mettre une nouvelle barrière :
#1 - car elle n'est pas alignée sur 2 cases complètes
#2 - parce qu'elle sort de la zone de jeu

Impossible de mettre une nouvelle barrière ici, sinon le joueur bleu est complètement bloqué

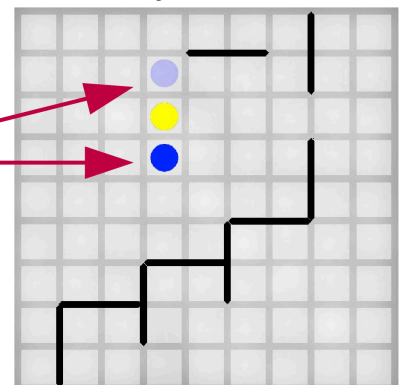


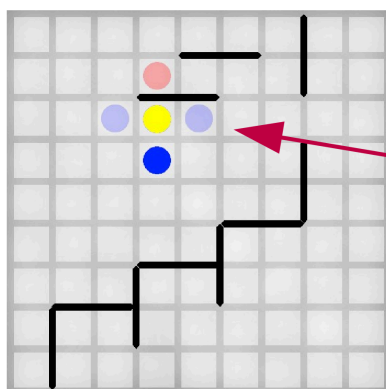
Le premier joueur à faire parvenir son pion sur n'importe quelle case qui se trouve sur le bord opposé à son point de départ est déclaré vainqueur.

Lorsque 2 pions sont côte à côte, il est possible pour l'un de "sauter" par dessus l'autre pour passer de l'autre côté en un seul coup. Ceci n'est possible QUE SI la case d'arrivée n'est pas déjà occupée par un autre joueur (cas d'une partie à 4 joueurs par exemple), ou bien que la case à sauter et la case d'arrivée ne soient pas séparées par une barrière.

Dans l'une des ces deux situations, et seulement dans ce cas là, il est possible pour le joueur actif de se déplacer en diagonale à côté du joueur adverse.

Le joueur bleu peut avancer de 2 cases vers le haut en 'sautant' par dessus le joueur jaune





Le joueur bleu ne peut pas avancer car la case d'arrivée est bloquée par une barrière. Il peut néanmoins se déplacer en diagonale.

Dans ce cas précis, si l'une des cases diagonales est séparée de la case du joueur jaune par une barrière, alors le déplacement diagonal ne pourra pas se faire non plus. Il faut voir ce "saut" comme si le joueur bleu avançait sur la case du joueur jaune, puis se déplaçait sur la case d'arrivée. Il effectue donc 2 déplacements consécutifs, si et seulement si ces 2 déplacements sont possibles (non entravés par une barrière, ou un 3ème joueur).

INFORMATIONS GENERALES

Taille de l'équipe

Ce projet est un travail d'équipe. Il est autorisé de se réunir en groupe de 4 ou 5 personnes.

Démarrage du projet

Vous obtiendrez de plus amples informations quant aux dates précises, les critères d'évaluation, le format du rapport associé à votre projet, les liens vers la personne chargée de vous suivre pendant ce projet, ..., quand le projet démarrera officiellement.

Jalons

En fonction de votre chargé de projet, et du sujet, vous aurez à fournir des éléments de progression à des dates spécifiques. Ces jalons dépendront du sujet du projet, de l'organisation de l'équipe, ainsi que du planning de votre chargé de projet. Le but principal est de prouver que votre équipe progresse dans le livrable du projet, et que tous les membres sont impliqués.

Rapport du projet

Un rapport écrit est requis, contenant une brève description de l'équipe et du sujet. Il décrira les différents problèmes rencontrés, les solutions apportées et les résultats. L'idée principale est de montrer comment l'équipe s'est organisée, et quel était le flux de travail appliqué pour atteindre les objectifs du cahier des charges. Le rapport du projet peut être rédigé en français.

Démonstration

Le jour de la présentation de votre projet, vous utiliserez la même version que celle fournie à votre chargé de projet (version finale livrée) même si vous avez ajouté des fonctionnalités ou corrigé des bugs entre temps. L'idéal est de déposer votre application sur un dépôt de code d'un outil de gestion de version.

Vous ferez votre démonstration, en fonction des exigences du cahier des charges de votre projet, et vous aurez à modifier votre application suite aux requêtes du jury.

De plus des questions supplémentaires pourront être posées afin d'évaluer votre connaissance de l'implémentation de votre application.

FONCTIONNALITES PRINCIPALES DU PROJET

- Le but principal du projet est de fournir une application fonctionnelle.
- Votre projet complet **devrait (dans l'idéal)** être stocké sur un dépôt git (ou un outil similaire) pour au moins trois raisons : éviter de perdre du travail tout au long du développement de votre application, être capable de travailler en équipe efficacement, et partager vos progrès de développement facilement avec votre chargé de projet. De plus il est **recommandé** de mettre en place un environnement de travail en équipe en utilisant divers outils pour cela (slack, Trello, Discord, ...).
- Tous les éléments de votre code **seront** écrits en langue anglaise.
- Votre application ne doit **jamais** s'interrompre de manière intempestive (crash), quelle que soit la raison. Toutes les erreurs doivent être gérées correctement. Il est préférable de d'avoir une application stable avec moins de fonctionnalités qu'une application contenant toutes les exigences du cahier des charges mais qui plante trop souvent.
- Vous **devez** fournir une **documentation** de l'ensemble de votre code en utilisant des outils dédiés tels que Doxygen ou Javadoc. Vous modifierez votre code en fonction. Votre livrable pourra être réutilisé pour régénérer la documentation comme vous l'aurez fait pour la livraison.
- Votre projet devra être codé en langage JAVA.
- Votre projet doit implémenter toutes les fonctionnalités listées en début de ce document.
- Votre projet devra être jouable, soit à la souris soit au clavier, au choix, pour déplacer le pion du joueur actif, placer une barrière, choisir son orientation,
- Le programme devra vérifier automatiquement qu'une barrière peut être posée ou non.
- Le programme devra vérifier automatiquement qu'un pion peut parvenir jusqu'au côté opposé : il est donc primordial d'implémenter un algorithme de recherche de chemin quel qu'il soit. Vu que la zone de jeu n'est pas très grande, n'importe quel algorithme, même s'il n'est pas optimisé en terme de complexité temporelle, devrait faire l'affaire.
- Votre projet doit pouvoir proposer une solution pour sauvegarder au moins une partie sur un fichier du disque dur, et la restaurer plus tard. Il faut donc un menu minimum pour réaliser ces fonctionnalités.

RESSOURCES UTILES

Github

- www.github.com
- <https://docs.github.com/en/get-started/quickstart/hello-world>

La sérialisation d'objets en JAVA

- <https://docs.oracle.com/javase/8/docs/technotes/guides/serialization/index.html>

Algorithmes de recherche de chemins

- https://fr.wikipedia.org/wiki/Recherche_de_chemin