

AI Project

Stratégie :

Nous avons voulu inculquer à notre IA les stratégies défensives et offensives de base, à savoir:

- Lorsqu'il y a déjà 3 pièces à caractère commun alignées, ne pas donner une pièce de ce caractère
- Lorsqu'il y a 3 pièces à caractère commun alignées sur le terrain et que nous avons en main une pièce à placer possédant ledit caractère, la placer afin de former un quarto

Implémentation :

Il était tout d'abord nécessaire, d'apprendre à notre intelligence à détecter si 3 pièces à caractère commun sont déjà alignées sur le terrain, c'est à ça que servent les fonctions `three<nomDuCaractère>`. Celles-ci fonctionnent à l'aide des expressions régulières (RegEx) : nous vérifions que le caractère que nous cherchons se trouve dans la pièce (ou pour Low et High, nous vérifions que la pièce corresponde à un pattern). Ceci combiné à une recherche horizontale, verticale et diagonale, nous permet de passer en revue tout le tableau et d'incrémenter un compteur lorsque la condition est remplie.

A l'aide de ces fonctions nous avons créé les fonctions `nextPieceToGive` et `nextPosition` qui devaient respectivement choisir la prochaine pièce à donner (une pièce qui ne permet pas de quarto) et choisir la position de la pièce qu'il faut placer (la 4ème position pour former le quarto ou une position aléatoire).

Cependant, bien que le code compile, lorsque l'on essaye de faire jouer l'intelligence artificielle le serveur nous renvoie l'erreur "A valid move must be a valid JSON string" devant "move" au tour de notre IA.

```
598 > def threeLow(self):=  
627  
628 > def threeHigh(self):=  
658  
659 > def threeLight(self):=  
687  
688 > def threeDark(self):=  
716  
717 > def threeFull(self):=  
745  
746 > def threeEmpty(self):=  
774  
775 > def threeRound(self):=  
803  
804 > def threeSquare(self):=
```