

Support de cours

Cours:

Initiation à la programmation (en Java)

Vidéo:

Init-JAVA-07-5-gagne-pt2

Concepts (extraits des sous-titres générés automatiquement) :

Méthode estcegagne. Ligne de la case de départ. Bord droit de la grille. Bas de la grille. Nombre de pions. Nombre de colonnes. Version finale de nos méthodes estcegagne. Nombre de lignes. Peine. Direction. Chose. Fois. Dernière vidéo. Répétitions. Colonne.



[vers la recherche de séquences vidéo](#)
(dans Initiation à la programmation (en Java).)



[vers la vidéo](#)

Center for Digital Education. Plus de matériel de soutien pédagogique ici :

<https://www.epfl.ch/education/educational-initiatives/cede/educational-technologies-gallery/boocs-en/>

Puissance 4 : méthodes `estCeGagne` et `compte`

(Partie 2)

Initiation à la programmation (Java)

Jamila Sam, Vincent Lepetit et Jean-Cédric Chappelier

...

notes

résumé

0m 0s





et même chose quand on part du bas de la grille, ce n'est pas la peine de chercher vers le bas.

notes

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

résumé

0m 37s



.....

.....

.....

.....

.....

Retour sur la méthode `estCeGagne`

```
if (couleurCase == couleurJoueur) {
    final int ligneMax = grille.length - 4;
    final int colonneMax = grille[ligne].length - 4;

    if (
        // en diagonale, vers le haut et la droite:
        (ligne >= 3 && colonne <= colonneMax &&
         compte(grille, ligne, colonne, -1, +1) >= 4) ||

        // horizontalement, vers la droite:
        (colonne <= grille[ligne].length - 4 &&
         compte(grille, ligne, colonne, 0, +1) >= 4) ||

        // en diagonale, vers le bas et la droite:
        (ligne <= grille.length - 4 && colonne <= grille[ligne].length - 4 &&
         compte(grille, ligne, colonne, +1, +1) >= 4) ||

        // verticalement, vers le bas:
        (ligne <= grille.length - 4 &&
         compte(grille, ligne, colonne, +1, 0) >= 4)
    ) {
        return true;
    }
}
```

Notre méthode `estCeGagne` fait donc plus d'opérations que nécessaire, et je vais donc modifier le code pour éviter de compter le nombre de pions quand on est sûr qu'il n'y aura pas au moins quatre pions d'alignés. Alors, ça va s'écrire de cette façon-ci. Quand on considère la direction en diagonale vers le haut et la droite, et bien pour qu'il y ait une chance d'avoir au moins quatre pions d'alignés, il faut que la ligne soit supérieure ou égale à trois, et que la colonne ne soit pas trop grande et je vous laisse vérifier quand vous faites, la colonne doit être inférieure au nombre de colonnes moins quatre. Quand on cherche vers la droite horizontalement, il suffit que la colonne ne soit pas trop grande et qu'elle soit inférieure, de même, au nombre de colonnes moins quatre. Quand on cherche en diagonale vers le bas et la droite, il faut que la ligne ne soit pas trop grande, et comme pour la colonne, il faut que la ligne soit inférieure au nombre de lignes moins quatre, et que la colonne ne soit pas trop grande et quand on compte le nombre de pions vers le bas, il suffit que la ligne soit inférieure au nombre de lignes moins quatre. Maintenant, vous pouvez remarquer que j'ai introduit plusieurs répétitions, puisque cette expression apparaît ici et ici, et cette expression apparaît également deux fois. Je vais donc introduire des constantes pour remplacer ces répétitions, constantes que je vais appeler logiquement `ligneMax` et `colonneMax`,

notes

résumé

0m 46s



et qui vont donc m'éviter ces répétitions. Voilà, cette fois-ci, c'était vraiment la version finale de nos méthodes estCeGagne et compte, il nous reste à gérer le cas où la grille devient pleine sans qu'aucun des deux joueurs n'ait gagné et c'est ce que vous allez voir dans la dernière vidéo. dans la dernière vidéo.

résumé

2m 25s



