Une image contenant texte, Police, Graphique, logo

Description générée automatiquementUne image contenant Police, Graphique, texte, logo

Description générée automatiquement

Rapport de Démineur



Karina Moreau & Fares Sfar

4A IMDS

2023-2024

Résumé

Dans le cadre de nos études d’Ingénierie Mathématique et Data Science, nous avons codé le jeu du Démineur en language Java. Le Démineur est un jeu qui nécessite à la fois la prise de décisions stratégiques et la logique pour éviter les mines et révéler toutes les cases sans risquer de déclencher une explosion.

L’objectif de ce rapport est de présenter notre travail, en argumentant les choix que nous étions amenés à faire pour obtenir des résultats convaincants.

Sommaire

Présentation du jeu

**Le menu : …**

**La partie :**

Le jeu affiche le nombre de Mines restantes

Le jeu commence par afficher le plateau avec toutes les cases cachées.

On propose au joueur de soit marquer la case en tappant « 1 » avec un drapeau « D », de découvrir la case en tappant « 2 » ou sinon, en tappant « 0 », on peut quitter la partie.

Si le joueur veut marquer ou découvrir une case alors, on lui demande les coordonnées.

Si on découvre une case, elle peut être chiffré, vide ou minée.

Si une case ne contient pas de mine, elle révèle un chiffre qui indique le nombre de mines adjacentes à cette case, horizontalement, verticalement et en diagonale.

Si une case ne contient pas de mine et n'a aucune mine adjacente, elle se révèle comme une case vide, et toutes les cases adjacentes vides sont également automatiquement révélées.

Si une case contient une mine, elle révèle une case « M » sur le plateau et donc, la partie se termine car, le joueur a perdu.

Le jeu se termine lorsque le joueur a découvert toutes les cases sans mines. Si le joueur clique sur une mine, le jeu se termine, et le joueur a perdu.

Programmation

Classe Partie

On applique l’instance de notre partie avec la méthode get et set

On implémente des méthodes de get pour le nombre des mines restantes, le nombre de mines initiales et le niveau car, ce sont des variables protégées qui retournent les nombres de mines restantes ou initiales ou le niveau de notre partie. De même pour les méthodes isResultat() et isEncours(), qui retournent le résultat de notre partie (si la partie est gagné ou perdu) ou retournent si la partie est en cours ou à l’arrêt.

Les méthodes marquerCase(Point pt), decouvrirCase(Point pt) et testerSiGagne() qui appelle ces mêmes fonctions dans la Classe Plateau

Il existe aussi des méthodes qui permettent d’augmenter ou diminuer à notre compteur de mines restantes.

La fonction perdre (la case découverte est minée) permet de dire que le résultat est perdu (=false) et que la partie n’est plus en cours (=false)

Dans cette classe, on affiche le nombre de mines restantes, le plateau implémenté dans la classe Plateau, on demande à l’usager de faire son choix dans le jeu : soit il marque une case, soit il découvre une case ou il quitte la partie. On fait une boucle switch de sorte à ce que : si le joueur veut la marquer ou la découvrir dans ce cas, on demande les coordonnées des cases à l’utilisateur et ainsi on la marque ou on la découvre. Puis on continu le procédé jusqu’à que le joueur décider de quitter.

Classe Plateau

On déclare une hashmap du plateau contenant ces points et ces cases, donc ces coordonnées

On applique la méthode des geteur et des seteur de la taille de notre plateau qui est déclaré en protected

La fonction testerSiGagne permet de dire que la partie est gagné si