

Programmation d'interfaces graphiques

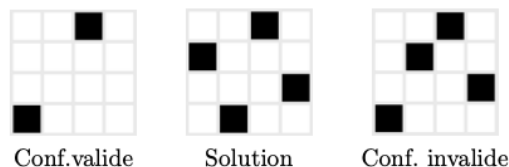
TP 4

2^{ème} année licence informatique et vidéoludisme

Université Paris 8

L'énigme des 8 reines

Le problème des 8 reines s'énonce ainsi : peut-on placer 8 reines sur un échiquier de 64 cases, de sorte qu'aucune reines ne soit menacée par une autre? Autrement dit, nous voulons placer 8 reines dans cet échiquier de sorte qu'il n'y ait jamais deux reines sur une même ligne, une même colonne, ou une même diagonale. Lorsqu'une reine est placée, la ligne, la colonne et les diagonales sur lesquelles elle se situe sont donc condamnées.



Dans ce TP, nous allons créer un plateau de jeu graphique qui proposera à un joueur de résoudre cette énigme. Le joueur pourra effectuer les deux actions suivantes sur le plateau :

- Cliquer sur une case libre pour y placer une reine.
- Cliquer sur une case occupée pour retirer la reine qui s'y trouve.

On ajoutera également au jeu deux boutons :

- Un bouton de validation, qui indique si la configuration actuelle du plateau est acceptable (c'est-à-dire qu'aucune des reines présentes n'en menace une autre, indépendamment du nombre de reines présentes).
- Un bouton de demande d'indice. Si la configuration actuelle peut être complétée en une solution, alors l'interface propose au joueur une case sur laquelle jouer. Sinon elle indique que le joueur est bloqué.

Dans tout ce TP, on complètera le squelette de fichier "*8Reines.py*" :

Question 1: Compléter le constructeur du plateau pour voir s'afficher l'échiquier et les deux boutons.

Question 2: A présent, on souhaite savoir pour chaque case s'il se trouve une reine. Combien y a-t-il d'état pour une case? Comment représenter cet état? Comment accéder à cet état?

Question 3: Un simple clic gauche sur une case du tableau place une dame s'il n'y en avait pas, ou l'enlève s'il y en avait une. Cela change la valeur de l'option *text* du bouton en un "X" pour représenter une case occupée, ou en une chaîne vide "" pour représenter une case libre. Compléter la méthode *clicGauche()* de la classe *Case*, pour obtenir ce comportement.

Question 4: Pour vérifier que les reines sont bien placées, certaines méthodes de cette partie devront parcourir les cases de l'échiquier. Dans quelle classe faudra-t-il les définir?

Question 5: Que font les méthodes *compteDiagonales()*, *compteAntidiagonale()*, *verifieDiagonales()*, *verifieAntidiagonales()* ?

Question 6: Écrire une méthode qui renvoie vrai si toutes les diagonales et toutes les anti-diagonales sont correctes.

Question 7: A présent, compléter le bouton validation, afin que le texte de celui-ci devienne rouge, si une diagonale, ou une antidiagonale contient deux reines, et reste vert sinon. Tester ensuite que cela fonctionne aussi bien pour le vert que pour le rouge.

Question 8: Écrire une méthode *compteLigne()* qui compte le nombre de reines présentes dans une ligne, et en déduire une méthode *verifieLignes()* qui renvoie *True* si et seulement si aucune ligne du plateau ne contient plus d'une reine. Enfin, procéder de même pour les colonnes.

Question 9: Compléter la méthode *verifieConfiguration()* pour vérifier aussi les lignes et les colonnes, et tester ensuite que cela fonctionne aussi bien pour le vert que pour le rouge.

Question 10: Programmer maintenant la méthode *solution()* pour compléter le comportement du bouton de demande d'indice. On va pour cela suivre les étapes suivantes:

- Vérifier la validité de la configuration actuelle
- Chercher la première ligne non occupée, et essayer les cases de cette ligne les unes après les autres.
- Compléter la méthode *indice()* pour que le bouton "indice" s'affiche en vert si la configuration peut être complétée, et en rouge sinon.

note: la méthode *solution()* renvoie un tuple (*admetSolution:bool*, *x:int*, *y:int*) où *admetSolution* est un booléen égal à *True* si la configuration actuelle permet de placer une autre reine, et *False* sinon. *x* et *y* sont les coordonnées de la case où on peut placer la reine.

Question 11: Compléter le code du bouton "indice" pour que dans le cas où une solution est trouvée, le jeu place pendant 1 seconde une reine de couleur verte dans la case où il serait judicieux d'en placer une.

Question 12: Ajouter un bouton "Réinitialiser" qui vide toutes les cases du tableau.