

IN200: programmation du jeu du taquin

Objectif général

Dans ce projet, on veut programmer le jeu solitaire du taquin. Il s'agit d'un puzzle constitué de 15 carrés numérotés de 1 à 15 qui peuvent coulisser horizontalement et verticalement à l'intérieur d'un cadre carré qui contient un emplacement vide. Un carré ne peut coulisser que si l'emplacement voisin dans la direction choisie est vide. L'objectif est de déplacer les carrés de manière à obtenir la configuration donnée à la figure [1](#)



FIGURE 1 – Taquin résolu (source Wikipédia)

En général, le jeu démarre à partir d'une configuration choisie au hasard comme illustré à la figure [2](#). Il faut noter que certaines configurations de départ n'admettent pas de solution.



FIGURE 2 – Taquin mélangé (source Wikipedia).

Travail demandé


Il faut fournir une interface graphique permettant de jouer au jeu du taquin. La configuration de départ doit être choisie au hasard, mais la case libre doit être en bas à droite. L'interface graphique est libre.

Votre programme doit également permettre de :

- sauvegarder une partie en cours puis de la recharger;
- revenir en arrière, c'est-à-dire annuler des déplacements de carrés.

Pour aller plus loin

Voici certaines fonctionnalités que vous pouvez programmer pour améliorer votre projet :

- ajouter la possibilité de pousser les carrés par bloc; sur la figure  on doit par exemple pouvoir déplacer les carrés 6, 4 et 14 ensemble vers la gauche;
- le déplacement des carrés doit se faire avec la souris, en faisant en sorte que le déplacement des carrés se fasse de manière continue, en suivant le mouvement de la souris;
- un critère simple, détaillé ci-après permet de déterminer si une configuration admet une solution ou pas; on demande alors de programmer une aide qui indique si la configuration du jeu est résolvable ou pas; de plus on améliorera le programme de manière à ne choisir comme configuration de départ uniquement des configurations résolvables choisies au hasard (en conservant le fait que le carré libre est en bas à droite);
- implémenter une IA pour résoudre le jeu.

Pour savoir si une configuration donnée admet une solution ou pas, il suffit de compter le nombre de permutations de carrés qui permettent de se ramener à la configuration où le taquin est résolu. Les configurations résolvables sont exactement celles pour lesquelles ce nombre est pair (la parité ne dépend pas de l'ordre dans lequel sont faites les permutations). En détail, une permutation entre 2 carrés consiste à échanger de manière fictive l'emplacement de ces carrés, étant compris que cette opération n'est pas une opération légale du jeu. Noter aussi qu'il n'est pas nécessaire que ces carrés soient voisins.