

2IA

Projet - sujet # 1

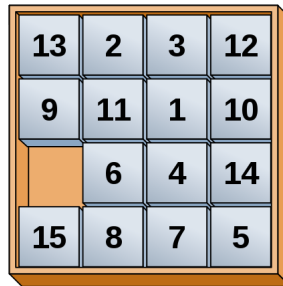
Remarque : Deux sujets de projet seront proposés. Le premier concerne l'implémentation de certaines stratégies de recherche dans le jeu du taquin. Le second portera sur l'apprentissage automatique et sera proposé une fois le cours sur l'apprentissage commencé. Il vous est demandé de réaliser un seul projet, à vous de choisir selon vos préférences.

Sujet 1 : Taquin - Implémentation de quelques stratégies de recherche

Selon Wikipédia

" Le taquin est un jeu solitaire en forme de damier créé vers 18701 aux États-Unis. Sa théorie mathématique a été publiée par l'American Journal of mathematics pure and applied² en 1879. En 1891, son invention fut revendiquée par Sam Loyd³, au moment où le jeu connaissait un engouement considérable, tant aux États-Unis qu'en Europe. Il est composé de 15 petits carreaux numérotés de 1 à 15 qui glissent dans un cadre prévu pour 16. Il consiste à remettre dans l'ordre les 15 carreaux à partir d'une configuration initiale quelconque.

Le principe a été étendu à toutes sortes d'autres jeux. La plupart sont à base de blocs rectangulaires plutôt que carrés, mais le but est toujours de disposer les blocs d'une façon déterminée par un nombre minimal de mouvements. Le Rubik's Cube est aujourd'hui considéré comme l'un des « descendants » du taquin."



L'objectif de ce projet est d'implémenter en langage Python quelques stratégies de recherche (non informée et informée) et de les appliquer au jeu du Taquin. Dans notre cas, on considère un puzzle de $k \times k$: un plateau carré de k colonnes et k lignes. Sur ce plateau sont disposées $k*k-1$ cases numérotées de 1 à $k*k-1$. Le but du jeu consiste à ordonner les cases en déplaçant la case vide dans les directions autorisées selon son emplacement actuel.

Les principales questions sont :

1. Représentation d'un état du jeu ;
2. Modèle de transition : A partir d'un état actuel, générer tous les états suivants possibles ;
3. Test si un état est final ;
4. Implémentation des stratégies de recherche (non informée et informée).

Le paramètre k doit être fourni par l'utilisateur ainsi que la stratégie à utiliser.

L'état initial est généré aléatoirement.

Le résultat affiché doit être la succession d'états depuis l'état initial jusqu'à l'état final.