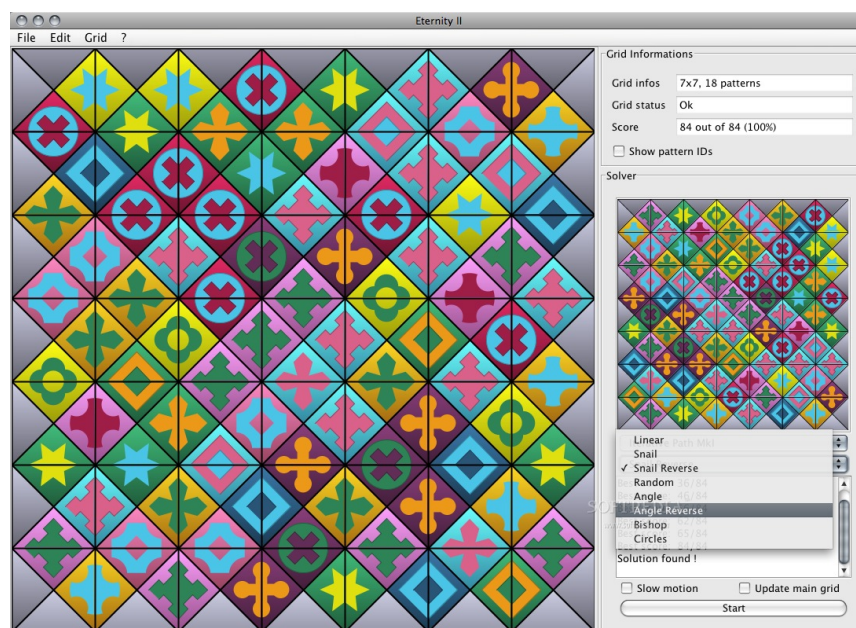


Dans ce projet vous devrez écrire un programme pour résoudre des puzzles du genre Eternity II (mais en plus petit sinon c'est trop dur).

Voici un exemple de puzzle de ce genre :



Il est constitué de tuiles carrées. Chaque tuile est divisé en 4 régions par ses diagonales. Chaque région a une couleur (dans l'image, on utilise couleur + forme pour la lisibilité, mais ce puzzle est isomorphe à un autre où on utiliserait plus de couleurs). Le but du puzzle est alors de mélanger les tuiles, puis à les replacer sur le damier de sorte que les couleurs de tuiles adjacentes se correspondent. Les bords sont gris pour éliminer des symétries : le bord du haut est gris foncé, les bords du côté sont gris normal, et le bord du bas est gris clair.

## 1 Résolution de puzzle

On considère un damier  $n \times m$  ainsi qu'un sac de  $nm$  tuiles (où chaque tuile a des couleurs affectées à ses 4 bords) à placer sur ces cases sous les conditions suivantes :

- à chaque case, on doit affecter une tuile différente (donc on utilisera toutes les tuiles).

- quand on affecte une tuile à une case, on peut utiliser n'importe quelle rotation de cette tuile.
- les bords extérieurs du damier doivent utiliser la couleur choisie pour les bords extérieurs (e.g. le gris). Aucun des bords intérieurs ne doit l'utiliser.
- les couleurs des bords de deux tuiles adjacentes doivent être identiques.

Votre tâche est d'écrire un programme qui à partir de la spécification de la taille du damier, et, étant donné un jeu de tuiles, calcule une solution du puzzle.

Le fichier `projet05.txt` contient une description du damier + jeu de tuiles correspondant à l'image. Le fichier contient une ligne :

```
| dimensions 7 x 7
```

indiquant les dimensions du damier (nombre de lignes, puis nombre de colonnes). Les autres lignes décrivent les tuiles et sont de la forme :

```
| N E S W
```

c'est à dire nord, est, sud, ouest. Chacun des éléments de ce quadruple est un symbole représentant une couleur. Vous devez lire le fichier `projet05.txt` pour en comprendre l'encodage ; vous devez aussi vérifier que j'ai correctement encodé l'exemple de l'image.

Le script python `projet05.py` illustre comment charger les données, et les affiche dans le format YAML. Il vous appartient de modifier ce script pour créer un/des fichier/s de données approprié pour Minizinc.

Pour tester votre modélisation, je vous recommande d'utiliser un puzzle plus petit.

La contrainte “élément” sera particulièrement utile : elle permettra d'avoir pour chaque tuile une variable représentant la rotation voulue (parmi les 4 possibles), et d'exprimer la configuration `n e s w` correspondante de la tuile comme un choix parmi 4 configurations possibles.