

Sudoku Solver

La plupart des gens ont déjà essayé de remplir une grille de sudoku. Cependant, la façon dont un cerveau humain résout un sudoku est bien différente de ce qu'un ordinateur peut faire. Notre projet s'intéresse à implémenter des algorithmes utilisant des techniques de plus en plus complexes, afin de résoudre des sudokus, eux-mêmes, de plus en plus complexes.

3 algorithmes ont été implémentés avec différents niveaux de complexité allant du plus basique au plus complexe. Les 2 premiers algorithmes se basent sur une méthode appelée : "backtracking". C'est une méthode de résolution assez connue en informatique. Le dernier solveur se base, lui, sur une résolution par contrainte liée à la théorie des graphes. Ces différents algorithmes ont été créés afin de pouvoir comparer la complexité et la rapidité d'exécution de ces différentes méthodes.

De plus, afin de tester une quantité astronomique de sudokus, 3 possibilités se sont démarquées : importer des grilles de sudoku, en créer soit même, et la création d'un générateur de sudoku. Afin de pouvoir générer un sudoku non-complet, il faut également utiliser les algorithmes décrit ci-dessus afin de vérifier si le sudoku généré peut être résolu.

Pour pouvoir présenter tout ceci au Printemps des Sciences, une interface graphique interactive a été mise à disposition afin de rendre ce projet beaucoup plus didactique.

The image shows the graphical user interface of the Sudoku Solver application. It consists of two main panels. The left panel is a dark-themed 25x25 grid with a grid of colored lines (red, green, blue, yellow) forming a 5x5 sub-grid structure. Overlaid on this grid are four menu items in white text boxes: "sudoku 25x25 b", "Algo par contraintes", "Résoudre", and "Retour au menu". The right panel is a white-themed 25x25 grid containing a partially filled 25x25 Sudoku puzzle. The numbers 1 through 25 are placed in various cells, with some cells being empty. The grid is also divided into 5x5 sub-grids by thicker lines.