

Rapport du projet Kakuro

Introduction

Je m'appelle Ayman Ghanmi, étudiant en L2-S4-TD1-TP2. Ce rapport présente mon projet intitulé "Programmation d'un solveur en C++ : Kakuro". L'objectif de ce projet est de développer un programme capable de résoudre automatiquement des grilles de Kakuro, un jeu de réflexion numérique proche des mots croisés.

Contexte et objectifs

Le projet s'inscrit dans le cadre de mes études en informatique et vise à approfondir mes compétences en programmation en C++. Le but est de concevoir un solveur performant capable de charger, analyser et résoudre des grilles de Kakuro, tout en respectant les contraintes de ce jeu.

Réalisation technique

Le programme est écrit en C++ et utilise des structures de données adaptées pour modéliser les grilles de Kakuro. Les principales fonctionnalités sont :

- Chargement d'une grille depuis un fichier texte (puzzle.txt)
- Modélisation des cellules avec une énumération pour distinguer les cellules noires, d'indices et jouables
- Vérification des contraintes de somme pour chaque ligne et colonne
- Résolution automatique de la grille en utilisant des algorithmes de recherche
- Affichage du résultat sous une forme claire et lisible

Structure du projet

Le projet est structuré en plusieurs parties :

- **src/** : Contient le code source (main.cpp), où l'algorithme de résolution est implémenté.
- **bin/** : Contient l'exécutable du programme.
- **resources/** : Contient la grille de test (puzzle.txt).
- **README.md** : Document d'accompagnement expliquant la compilation et l'exécution.
- **Makefile** : Script de compilation.

Utilisation

Pour compiler le projet, utilisez la commande suivante :

```
make
```

Pour exécuter le programme :

```
./bin/kakuro
```

Le programme charge automatiquement la grille depuis le fichier puzzle.txt et affiche la solution.

Conclusion

Ce projet m'a permis de mettre en pratique mes compétences en algorithmique et en gestion de fichiers en C++. Le solveur est capable de traiter efficacement des grilles de petite et moyenne taille. Des améliorations futures incluraient l'optimisation des algorithmes pour résoudre des grilles plus complexes et l'ajout d'une interface graphique.