

Projet Informatique

But du projet

L'objectif de ce projet informatique est d'implémenter une application permettant de résoudre une grille de Sudoku de niveau facile, moyen, difficile, diabolique. Le programme doit afficher la grille résultat d'une grille donnée et la sauvegarder dans un fichier texte.

Description du sujet

Historique – Principe du jeu

Le Sudoku est un jeu très populaire depuis quelques années. Il se présente généralement sous la forme d'une grille de 9 cases sur 9 cases et est inspiré du problème du carré latin et ainsi que du problème des 36 officiers de Leonhard Euler. Son origine remonterait à 1979 ...

Le but du jeu est de remplir la grille avec des chiffres 1 à 9, de telle sorte qu'un même chiffre ne se trouve jamais plus d'une fois sur une même ligne, dans une même colonne ou dans une même sous-grille de taille 3*3. Quelques cases sont en principe déjà remplies de manière à simplifier la résolution de grille.

Un exemple

La grille suivante est une grille de niveau « facile ». Elle peut vous permettre de découvrir le jeu si vous ne le connaissez pas, ou à vous exercer en réfléchissant à une façon de programmer les techniques de résolution que vous employez pour la résoudre.

6	4			1	3	9		
1				2	6	4		
	2	9		4	5	7		
		2				8	3	
8	6			3	7		1	9
7			2		9			
		1	3			6	9	
9	3	6	4		8		2	
		5						

La dimension d'une grille de Sudoku n'est pas toujours de 9*9 et vos programmes doivent pouvoir traiter tous les cas spécifiés (ainsi donc prévoir une constante MAX pour les tailles et introduire une taille, par exemple 9). Une grille peut aussi être composée de caractères à la place des chiffres.

Quelques consignes

Ce travail étant un projet, on vous laisse volontairement assez libre sur la direction à suivre. L'objectif ici n'est pas de récupérer une même version d'un programme, mais bien de vous obliger à vous organiser de manière à construire votre programme progressivement.

La principale consigne est de ne pas essayer de réaliser directement un programme pour résoudre une grille de Sudoku donnée. Il s'agit d'un projet, et donc d'un travail plus conséquent qu'un simple TP. Il est donc conseillé d'organiser votre travail selon un plan qui peut s'approcher du modèle suivant:

1. « Découverte » du Sudoku: principe, résolution de l'exemple, ...
2. Recherche des méthodes « classiques » à mettre en oeuvre pour résoudre une grille, comment définir la difficulté d'une grille, ...
3. Représentation des données, lecture d'une grille (saisie clavier, fichier texte, ...), principe des

- algorithmes à mettre en oeuvre.
4. Développement et mise au point du programme.
 5. Préparation de l'évaluation.

Pour répondre aux points 1 et 2, Internet permet de découvrir de très nombreux sites dédiés au Sudoku (Wikipédia a sa page spéciale Sudoku par exemple). Lorsque vous trouvez un site qui vous semble intéressant, vous pouvez le préciser aux autres membres du groupe.

Attention toutefois à ne pas tomber dans un piège tentant: récupérer un code source et se l'octroyer. Il est certain que ce genre de méthode ne pourrait aboutir qu'à une mauvaise note car il ne serait pas difficile de remarquer que le code en question n'est pas de vous ... prudence !!

Avancez par étapes pour mettre au point votre algorithme. Par exemple la phase de saisie d'une grille doit être au point avant de vouloir la résoudre. De même lors de la programmation des méthodes de résolution, commencez par implémenter des méthodes permettant de résoudre des grilles faciles avant de mettre en oeuvre des techniques de résolution plus évoluées, ...

Ainsi donc une grille sera saisie dans un fichier texte sous éditeur et sera lue et transformée pour être rangée dans la structure de données choisie.

*La grille se représente tout naturellement par un tableau à deux dimensions qui comprendra aussi des compteurs par ligne, colonne et région: une grille $MAX*MAX$ nécessitera un tableau $(MAX+2)*(MAX+1)$ pour inclure les compteurs.*

Les méthodes de résolution nécessitent ou impliquent souvent de connaître les chiffres possibles (et leur nombre) pour chaque case. Il est donc judicieux d'initialiser un cube qui contiendra la liste des chiffres possibles pour chaque case avec des compteurs par case, ligne, colonne et région. Ce cube sera de dimension $(MAX+2)(MAX+1) * (MAX+1)$. Le plan 0 contiendra les compteurs de nombre de chiffres par case; le plan 1 contiendra des 0/1, c-à-d la possibilité de ce chiffre 1 pour chaque case, entourés de compteurs par ligne, colonne et région; idem pour les plans 2,3,4,...*

Il faudra initialiser le cube, puis écrire des modules de mise à jour du cube pour une valeur de la grille et mettre à jour le cube en fonction de la grille initiale.

Ajouter des commentaires dans votre programme de manière à en faciliter la compréhension. Cela est valable pour l'enseignant qui va récupérer votre code pour l'évaluer, mais aussi pour vous si vous êtes amenés à vouloir reprendre ce code par la suite.

Pour les plus avancés, vous pouvez essayer d'implémenter une version graphique permettant d'améliorer l'affichage de la grille et de la solution. Pour cela vous pouvez découvrir la bibliothèque graphique gtk ou sdl si votre programme est réalisé en C par exemple.

Une autre possibilité est de s'intéresser à la génération de grilles de Sudoku.

Résultats attendus - évaluation

L'évaluation de votre travail se fera grâce à la remise d'un rapport (~10 pages). Ce rapport doit entre autres contenir une bibliographie référant les adresses de sites Internet qui vous ont été utiles. Finalement la justesse de votre programme sera aussi prise en compte, par ex envoi de votre code source (uniquement les fichiers sources .h, .c).