



SUDOKU

CAHIER DES CHARGES

SUDOKU C'EST PAS POUR LES
MATHEUX



PARTICIPANTS :

Nangninta CLEBIEN 38017226

Ubald KELOUTE 38016940

SOMMAIRE

I - OBJECTIF DE L'APPLICATION

II - LES SPECIFICATIONS FONCTIONNELLES

III - LES SPECIFICATIONS TECHNIQUES

IV - CONCLUSION

I. LES OBJECTIFS DE L'APPLICATION

Créer un programme en C qui permet de générer des grilles de sudoku (9 x 9) avec plusieurs niveaux de difficulté. Et un solveur pour ces grilles. Ainsi toute grille générée devrait être solvable et le solveur devrait pouvoir le résoudre, avec des niveaux de difficultés différents. Permettre aussi à un utilisateur

II. LES SPECIFICATIONS FONCTIONNELLES

Les règles du sudoku sont très simples. Un sudoku classique contient neuf lignes et neuf colonnes, donc 81 cases au total. Le but du jeu est de remplir ces cases avec des chiffres allant de 1 à 9 en veillant toujours à ce qu'un même chiffre ne figure qu'une seule fois par colonne, une seule fois par ligne, et une seule fois par carré de neuf cases.

- Au début du jeu, des chiffres sont déjà placés dans la grille et le nombre de chiffre varie en fonction du niveau de difficulté du jeu et il vous reste à trouver les autres en respectant le principe de la ligne, de la colonne et du bloc (carré de 9 cases).
- Une grille initiale de sudoku correctement constituée ne peut aboutir qu'à une et une seule solution. Pour trouver les chiffres manquants, tout est une question de logique et d'observation.

- Dans notre programme il y'a plusieurs fonctionnalités qu'on aurait souhaité implémenter mais qu'on n'a pas pu faire. D'une part à cause du manque de temps et d'autre part à cause des limitations du langage C. Nous Parlons notamment de la partie interface graphique.

- Nous allons faire une présentation générale du programme tout en faisant le tour des fonctionnalités
 - Nous avons tout d'abord le menu principal qui nous permet de choisir ce qu'on souhaite faire
- 1- pour nous permettre de connaître les règles du jeu, 2- Nous permet de jouer, 3- nous permet d'afficher des grilles et leur solution, 4- nous permet de quitter

C:\Ubald\Nanterre Université\L3\Algo\Projet_KELOUTE_CLEBIEN\projet_harmo.exe

```
-----
BIENVENUE SUR LE JEU SUDOKU  SOUHAITEZ VOUS CONTINUER?      |
-----
1:  -----A PROPOS DU JEU-----|
-----
2:  -----JE SOUHAITE JOUER-----|
-----
3:  -----GENERATEUR ET SOLVEUR DES GRILLES-----|
-----
4:  -----JE SUIS PAS INTERESSE-----|
-----
-----VOTRE CHOIX ?-----
```

- Si nous choisissons 1, Nous avons le principe du jeu et les différentes options qui nous permettent de rentrer au menu principal, de jouer ou de quitter.

C:\Ubald\Nanterre Université\L3\Algo\Projet_KELOUTE_CLEBIEN\projet_harmo.exe

```
le principe du jeu est tres simple
1: Le but du jeu est de remplir ces cases avec des chiffres allant de 1 jusqu'a 9'
2: Chaque chiffre figure une fois sur une ligne, une fois sur une colonne et une fois dans le carré
3: La ligne est un ensemble de neuf cases/cellules disposées horizontalement
4: La colonne est un ensemble de neuf cases disposées verticalement
5: Le carré est un ensemble de neuf cases disposées en carré de 3x3 cases, la grille étant composée de neuf de ces carrés
Attention vous avez un nombre limité de tentatives faut être tactique
Souhaitez vous jouer ?
1:OUI
2:REVENIR AU MENU PRINCIPAL
3:JE SOUHAITE QUITTER
```

- Si à partir du menu nous choisissons l'option 3 à partir du menu, nous avons différentes grilles qui sont générés et leur solution. Exemple ci-dessous

C:\Ubald\Nanterre U

```
3
0 0 3|4 0 6|7 0 9|
0 0 6|7 0 0|0 0 3|
7 8 0|1 2 0|4 5 0|
-----
```

```
2 1 0|3 6 5|0 9 0|
3 6 5|8 9 7|2 0 4|
0 9 7|0 1 4|3 6 5|
-----
```

```
0 0 0|6 0 2|9 0 8|
0 0 0|9 7 0|5 3 1|
9 0 8|5 3 1|6 4 0|
-----
```

```
1 2 3|4 5 6|7 8 9|
4 5 6|7 8 9|1 2 3|
7 8 9|1 2 3|4 5 6|
-----
```

```
2 1 4|3 6 5|8 9 7|
3 6 5|8 9 7|2 1 4|
8 9 7|2 1 4|3 6 5|
-----
```

```
5 3 1|6 4 2|9 7 8|
6 4 2|9 7 8|5 3 1|
9 7 8|5 3 1|6 4 2|
-----
```

```
1 0 7|2 3 8|5 6 9|
0 0 8|1 0 0|0 4 7|
3 6 0|4 5 7|1 2 0|
-----
```

```
4 7 1|3 8 2|0 9 5|
5 8 2|6 9 1|4 7 3|
6 9 3|0 7 4|2 8 1|
-----
```

```
0 0 4|8 2 3|9 5 6|
0 0 5|9 1 0|7 3 4|
9 3 6|7 4 5|8 1 2|
-----
```

```
1 4 7|2 3 8|5 6 9|
2 5 8|1 6 9|3 4 7|
3 6 9|4 5 7|1 2 8|
-----
```

```
4 7 1|3 8 2|6 9 5|
5 8 2|6 9 1|4 7 3|
6 9 3|5 7 4|2 8 1|
-----
```

```
7 1 4|8 2 3|9 5 6|
8 2 5|9 1 6|7 3 4|
9 3 6|7 4 5|8 1 2|
-----
```

C:\Ubald\Nanterre U

```
0 2 0|4 5 8|3 6 9|
0 5 0|1 6 9|0 2 0|
0 0 0|0 3 7|0 0 0|
```

```
-----
0 0 4|3 0 5|8 9 0|
0 0 3|0 9 6|2 1 0|
0 0 1|0 0 0|7 5 0|
```

```
-----
0 1 0|6 0 3|0 7 0|
0 0 5|0 0 0|0 0 2|
0 4 0|5 0 0|1 3 0|
```

```
-----
1 2 7|4 5 8|3 6 9|
3 5 8|1 6 9|4 2 7|
4 6 9|2 3 7|5 8 1|
```

```
-----
2 7 4|3 1 5|8 9 6|
5 8 3|7 9 6|2 1 4|
6 9 1|8 2 4|7 5 3|
```

```
-----
8 1 2|6 4 3|9 7 5|
7 3 5|9 8 1|6 4 2|
9 4 6|5 7 2|1 3 8|
```

```
-----
0 4 0|2 5 0|0 3 9|
0 0 0|1 3 0|0 0 0|
0 0 0|0 0 8|0 0 0|
```

```
-----
0 0 2|0 0 1|5 9 0|
0 0 0|0 6 2|4 8 0|
0 0 3|0 0 0|0 0 0|
```

```
-----
0 1 0|8 0 0|0 6 0|
0 0 0|0 0 0|0 0 0|
0 0 0|6 0 0|0 1 0|
```

```
-----
1 4 7|2 5 6|8 3 9|
2 5 8|1 3 9|6 4 7|
3 6 9|4 7 8|1 5 2|
```

```
-----
4 7 2|3 8 1|5 9 6|
5 9 1|7 6 2|4 8 3|
6 8 3|5 9 4|7 2 1|
```

```
-----
7 1 4|8 2 3|9 6 5|
8 2 6|9 1 5|3 7 4|
9 3 5|6 4 7|2 1 8|
```

-Intéressons-nous maintenant à la partie qui permet à un utilisateur de jouer. Si nous choisissons l'option 2 à partir du menu principal on arrive à choisir le niveau de difficulté. En fonction du niveau de difficulté choisi, nous avons un certain nombre d'itération au-delà duquel on ne pourra plus jouer. Une fois la grille pleine ça veut dire que nous avons bien et bel trouvé la solution car avant de jouer une valeur les vérifications sont effectués (si la valeur est bien valide dans cette ligne, colonne et carré).

C:\Ubald\Nanterre Universit \L3\Algo\Projet_KELOUTE_CLEBIEN\projet_harmo.exe

```

BIENVENUE SUR LE JEU SUDOKU  SOUHAITEZ VOUS CONTINUER?      |
-----
1:  -----A PROPOS DU JEU-----|
-----
2:-----JE SOUHAITE JOUER-----|
-----
3:-----GENERATEUR ET SOLVEUR DES GRILLES-----|
-----
4:-----JE SUIS PAS INTERESSE-----|
-----
-----VOTRE CHOIX ?-----
2
Veuillez choisir votre niveau de difficult  en entrant un nombre
1 : Niveau Facile
2 : Niveau intermediaire
3 : Niveau difficile

```

-Attention nous ne pouvons effacer que les valeurs qui ont  t  mis par nous et non ceux qui nous sont fourni au d part. On ne peut pas effacer une case qui est d j  vide. Nous allons voir des illustrations ci-dessous

-Nous choisissons le niveau de difficult 

C:\Ubald\Nanterre Universi

```

La grille a resoudre est
1 4 7|0 0 0|5 6 9|
2 0 8|0 6 9|0 4 7|
3 6 0|4 5 7|0 2 0|
-----
4 0 1|0 8 0|6 0 0|
5 8 2|0 9 1|0 7 3|
6 9 3|5 0 0|2 8 0|
-----
7 1 4|8 0 3|0 5 0|
8 0 0|0 1 0|7 3 4|
9 3 0|0 4 0|8 1 0|
-----

1 : jouer une valeur
2: effacer une valeur

```

-Un message d'erreur affich  l'orsque nous essayons de supprimer une case que nous n'avons pas le droit de supprimer

```
C:\Ubaldo\Nanterre Universit  \L3\Algo\Projet_KELOU

La grille a resoudre est
1 4 7|0 0 0|5 6 9|
2 0 8|0 6 9|0 4 7|
3 6 0|4 5 7|0 2 0|
-----
4 0 1|0 8 0|6 0 0|
5 8 2|0 9 1|0 7 3|
6 9 3|5 0 0|2 8 0|
-----
7 1 4|8 0 3|0 5 0|
8 0 0|0 1 0|7 3 4|
9 3 0|0 4 0|8 1 0|
-----

1 : jouer une valeur
2: effacer une valeur
2

Veuillez entrer l'indice de la ligne    effacer
0

Veuillez entrer l'indice de la colonne    effacer
1

Case deja vide ou non supprimable
1 4 7|0 0 0|5 6 9|
2 0 8|0 6 9|0 4 7|
3 6 0|4 5 7|0 2 0|
-----
4 0 1|0 8 0|6 0 0|
5 8 2|0 9 1|0 7 3|
6 9 3|5 0 0|2 8 0|
-----
7 1 4|8 0 3|0 5 0|
8 0 0|0 1 0|7 3 4|
9 3 0|0 4 0|8 1 0|
-----

1 : jouer une valeur
2: effacer une valeur
```

-Nous jouons une valeur dans une case de notre choix


```
C:\Ubald\Nanterre Universit  \L3\Algo\Projet_KELOUTE_CLEBIEN\proj

-----
7 1 4|8 0 3|0 5 0|
8 0 0|0 1 0|7 3 4|
9 3 0|0 4 0|8 1 0|
-----

1 : jouer une valeur
2: effacer une valeur
1

Veuillez entrer l'indice de la ligne ou faire l'affectation
1

Veuillez entrer l'indice de la colonne ou faire l'affectation
1

Veuillez entrer la valeur a affecter
5
1 4 7|0 0 0|5 6 9|
2 5 8|0 6 9|0 4 7|
3 6 0|4 5 7|0 2 0|
-----
4 0 1|0 8 0|6 0 0|
5 8 2|0 9 1|0 7 3|
6 9 3|5 0 0|2 8 0|
-----
7 1 4|8 0 3|0 5 0|
8 0 0|0 1 0|7 3 4|
9 3 0|0 4 0|8 1 0|
-----

1 : jouer une valeur
2: effacer une valeur
```

-Nous pouvons    pr  sent effacer une valeur que nous avons-nous m  me jou  s.


```

C:\Ubald\Nanterre Universit \L3\Algo\Projet_KELOUTE_CLEBIEN\pro
1 4|8 0 3|0 5 0|
0 0|0 1 0|7 3 4|
3 0|0 4 0|8 1 0|
-----

1 : jouer une valeur
2: effacer une valeur

Veuillez entrer l'indice de la ligne ou faire l'affectation

Veuillez entrer l'indice de la colonne ou faire l'affectation

Veuillez entrer la valeur a affecter

aleur non valide ou case non vide veuillez reessayer
4 7|0 0 0|5 6 9|
0 8|0 6 9|0 4 7|
6 0|4 5 7|0 2 0|
-----
0 1|0 8 0|6 0 0|
8 2|0 9 1|0 7 3|
9 3|5 0 0|2 8 0|
-----
1 4|8 0 3|0 5 0|
0 0|0 1 0|7 3 4|
3 0|0 4 0|8 1 0|
-----

1 : jouer une valeur
2: effacer une valeur

```

III. LES SPECIFICATIONS TECHNIQUES

Pour mettre en place notre programme on utilisera l'environnement de d veloppement **Code block**, on d finira les fonctionnalit s suivantes qui sont les prototypes qui nous permettrons de faire fonctionner notre programme :

- Une structure grille : qui nous permet de g rer nos grilles
- int indicePosition() : la fonction qui stocke les indices des cases qu'on a le droit de supprimer
- int existePos(int i, int j) : V rifie si la position prise en param tre est modifiable car on ne peut supprimer ou modifier que des valeurs qu'on a ajout  nous m me
- int grillePleine() : V rifie si une grille est pleine c'est- -dire si toutes ses valeurs sont non nulle

- int menu() : Le menu de notre application qui offre des options
- void effacer() : Permet d'effacer une valeur que le joueur a rentré.
- void ajoutValeur() : Permet d'ajouter une valeur à une position du tableau
- void ajoutGrille(grille *t) : ajoute une grille dans notre tableau de grille
- void afficher() : affiche un sous menu de notre programme qui offre des options à l'utilisateur
- int equalsGrille(grille grille2, grille grille1) : Compare deux grilles
- void generateur(grille *t, int n) : Notre générateur de grille
- void affichage(int grille[9][9]) : affiche une grille
- int resolution(int ligne, int colonne, int grille[9][9]) : Résout une grille c'est le solveur
- void differentesGrilles() : génères différentes grilles et les solutions respectives
- int absentSurLigne (int valeur, int ligne, int colonne, int grille[9][9]) : Vérifie si une valeur est présente sur une ligne
- int absentSurColonne (int valeur, int ligne, int colonne, int grille[9][9]) : Vérifie si une valeur est présente sur une colonne
- int absentSurBloc (int valeur, int ligne, int colonne, int grille[9][9]) : Vérifie si une valeur est présente dans un bloc 3x3
- int jeu() : Permet d'afficher l'interface de jeu pour l'utilisateur

IV. CONCLUSION

Ce projet a été très intéressant car il a développé en nous l'esprit d'équipe qui est un atout majeur dont tout développeur doit acquérir, en outre ça été l'occasion pour nous de nous surpasser et d'être créatif.