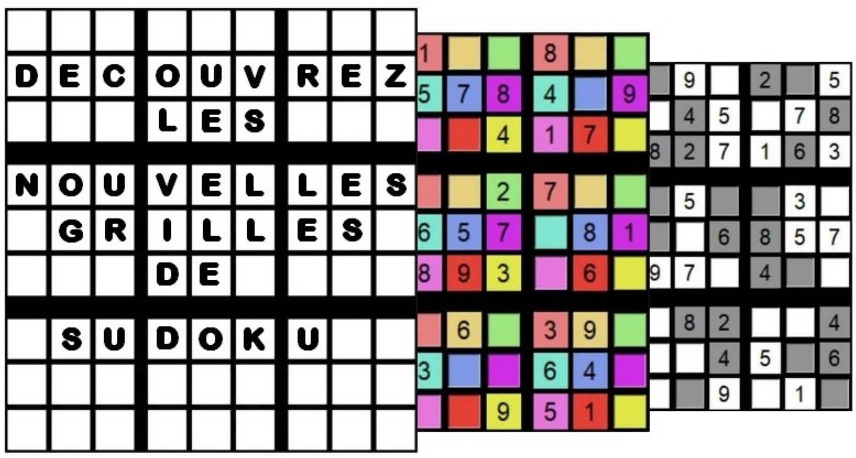
Licence 3 Mathématiques-Informatique

Rapport de projet

Génération de grilles Sudoku et vérification de l'unicité de la solution



M.Akim SAADI
M.Lucas ARCAS
M.Zakari IKHOU
M.Yaniss KHELFI AHMED
M.Guillaume NGUYEN THI

Tuteur : M.Benjamin MONMEGE

2019 — 2020

Table des matières

Génération de grilles Sudoku et vérification de l'unicité de la solution.

Akim SAADI[1], Lucas ARCAS[2], Zakari IKHOU[3], Yaniss KHELFI AHMED[4], and Guillaume NGUYEN THI[5]

1 **Aix-Marseille Université, France,** `akim.saadi@etu.univ-amu.fr`

2 **Aix-Marseille Université, France,** `lucas.arcas@etu.univ-amu.fr`

3 **Aix-Marseille Université, France,** `zakari.ikhou@etu.univ-amu.fr`

4 **Aix-Marseille Université, France,** `yaniss.Khelifi-ahmed@etu.univ-amu.fr`

5 **Aix-Marseille Université, France,** `guillaume.nguyen-thi@etu.univ-amu.fr`

Résumé

La demande croissante de grilles de Sudoku suite à la popularisation de ce jeu de logique à la fin des années 90 a poussé à informatiser leur génération. Pour répondre à cette demande il a fallu développer des algorithmes pour permettre d'assurer une solution unique tout en proposant une difficulté raisonnable en jouant sur le nombre d'indices de départ. Puis la curiosité à pousser les chercheurs à connaître et prouver le nombre minimum d'indices permettant de maintenir une solution unique, ainsi que la création de variantes avec des règles supplémentaires, des règles modifiées ou encore pleins d'autres extravagantes variantes.

Mots clés Région ; Solution unique ; Sudoku ; Génération ; Variantes ; Algorithme

1 Introduction.

1.1 L'histoire du Sudoku.

Le Sudoku tel qu'on le connaît aujourd'hui est apparu en 1986 dans une revue de jeux japonaise proposant à ses lecteurs des jeux de logique et de réflexion, sous le nom Sudoku signifiant 'simple chiffre'. Il semblerait que celui-ci dans sa version 9x9 ait été crée par Howard Garns, un ancien architecte de 74 ans vivant aux états unis en 1979, inspiré en partie d'un type de puzzle beaucoup plus ancien : le carré latin. C'est un jeu de logique utilisant des combinaisons de chiffre où le joueur cherche à remplir une grille, tout en respectant des règles, chaque chiffre ne devant apparaître qu'une fois par ligne, colonne et région (ou zone), tout comme le Sudoku.

Revenons au Sudoku, à première vue, c'est un jeu simple où il suffit de remplir des cases avec des chiffres, mais en fonction de la difficulté de la grille, le joueur est appelé à utiliser des méthodes de raisonnement complexes faisant appel à des principes découlant des règles du Sudoku mais n'étant pas évidentes. Une difficulté était que les journaux voulant proposer à leurs lecteurs des grilles de Sudoku à résoudre avaient besoin que la solution soit unique, pour que l'éditeur puisse par la suite proposer aux lecteurs une solution à la grille. Il a donc fallu générer une grille complète, puis ensuite retirer des chiffres tout en maintenant une solution unique, la question fut alors jusqu'à combien d'indices était il possible de vider la grille tout en étant sur que celle ci possédait une solution unique.

Nous verrons dans un premier temps comment il a été prouvé qu'aucune grille ne comportant 16 indices ne possède une solution unique, puis nous verrons comment certains algorithmes utilisés fonctionnent, on expliquera alors quelques variantes de Sudoku qui existent ou que nous avons implémentées et enfin on parlera un peu du code source.

2 La nécessité d'avoir au moins 17 indices dans la grille à résoudre.

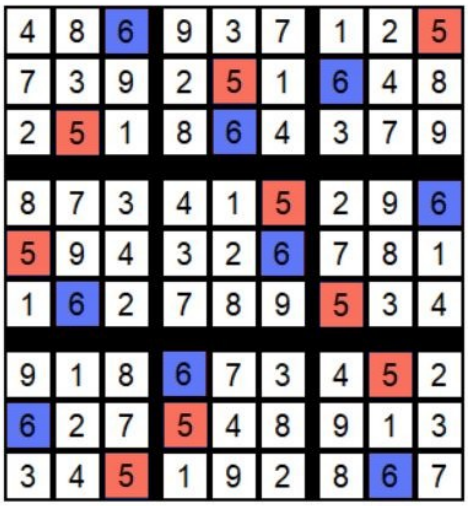
6 670 903 752 021 072 936 960, ou environ 10^{21}, c'est le nombre de grille complètes de Sudoku classique, 9x9 qui existent, et il a été récemment prouvé que le nombre minimum d'indices (nombre de chiffres dans la grille à résoudre) afin que la solution soit unique est 17[1], un grand nombre de mathématiciens avaient auparavant essayé de le prouver par une preuve mathématique 'classique', mais la méthode utilisée est un mélange de raisonnement mathématique et d'une méthode dite 'bruteforce', où grâce à de puissants ordinateurs on peut tester un grand nombre de cas[2] afin de vérifier toutes les issues possibles et donc en montrant de façon exhaustive que aucune grille à 16 indices ne possède une solution unique.

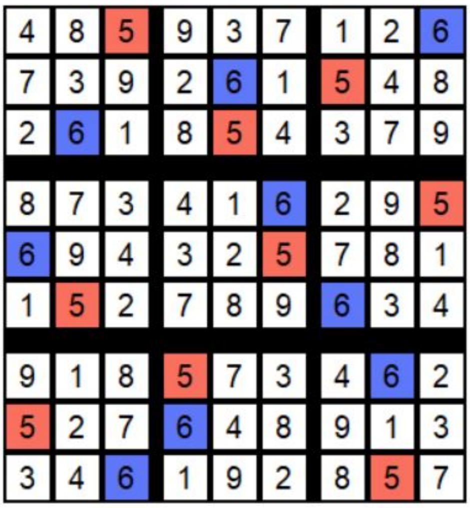
2.1 Comment il a été prouvé que 16 indices ne permettent pas d'avoir une solution unique.

Comme nous l'avons vu plus tôt, il y a en tout environ 10^{21} grilles possibles, or pour chaque grille il aurait fallu tester si pour chaque combinaison possible de 16 indices celle ci possède une solution unique, cela donne 10^{16} combinaisons possible de choisir 16 cases parmi 81, cela équivaut a plus de 300 000 ans en temps de compilation. Il a donc fallu réduire ce nombre de grilles à tester. En effet, il se trouve que certaines grilles sont semblables, nous allons voir comment réduire drastiquement le nombre de grilles à tester, puisque si une grille possède une solution unique, toutes les grilles semblables à celle ci auront elles aussi une seule solution.

2.2 L'idée de 'grilles semblables'.

Alors, qu'est ce qu'une grille semblable ? Pour chaque grille, il y a une multitude de grilles semblables, par exemple si on substitut les 6 par des 5 et les 5 par des 6, la grille garde les mêmes propriétés, si ce n'est que les 5 sont devenu des 6 et vice versa, et si la première grille avait une solution unique, alors la seconde aura elle aussi une solution unique.





De la même manière, il est possible d'intervertir la première ligne avec la seconde ou la troisième, la grille gardera ses propriétés d'unicité des chiffres par ligne, colonne et région et de plus si la grille avant permutation avait une solution unique, alors la seconde elle aussi aura une solution unique, il est possible de faire la même chose pour les lignes de 4 à 6, et de 7 à 9, ou encore pour les colonnes.

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

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

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

Une autre situation qui nous ramène a des grilles semblables est le cas suivant :

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

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

Si aucune de ces 4 cases n'est un indice donné au départ, cela nous amènerait à plusieurs solutions possibles, il est donc forcé que dans les grilles comprenant cette formation, au moins un des 4 chiffres soit un indice, réduisant ainsi encore le nombre de grilles à tester.

Même avec ces simplifications permettant de diminuer le nombre de grilles à tester jusqu'à n'avoir "que" 5 472 730 538 grilles, ce qui à quand même pris plus de 7 millions d'heures de compilation, ils ont commencer en janvier 2011 et on finit en décembre de la même année, prouvant ainsi qu'il n'existe pas de grille à 16 indices ayant une solution unique.

3 Fonctionnement des algorithmes.

3.1 Algorithme de vérification de l'unicité de la solution.

Comme nous l'avons vu auparavant[3], il est important que la grille créée possède une solution unique, nous allons voir comment arriver à une grille partiellement vidée, tout en étant certain qu'elle vérifie cette propriété d'avoir une solution unique.

Il existe bien sûr plusieurs manières de tester l'unicité d'une grille Sudoku, nous avons utilisé une méthode à base de backtracking fonctionnant de la manière suivante :

1. L'algorithme démarre d'une grille déjà rempli.

2. Il vide une case de la grille à une position (ligne, colonne) aléatoire.

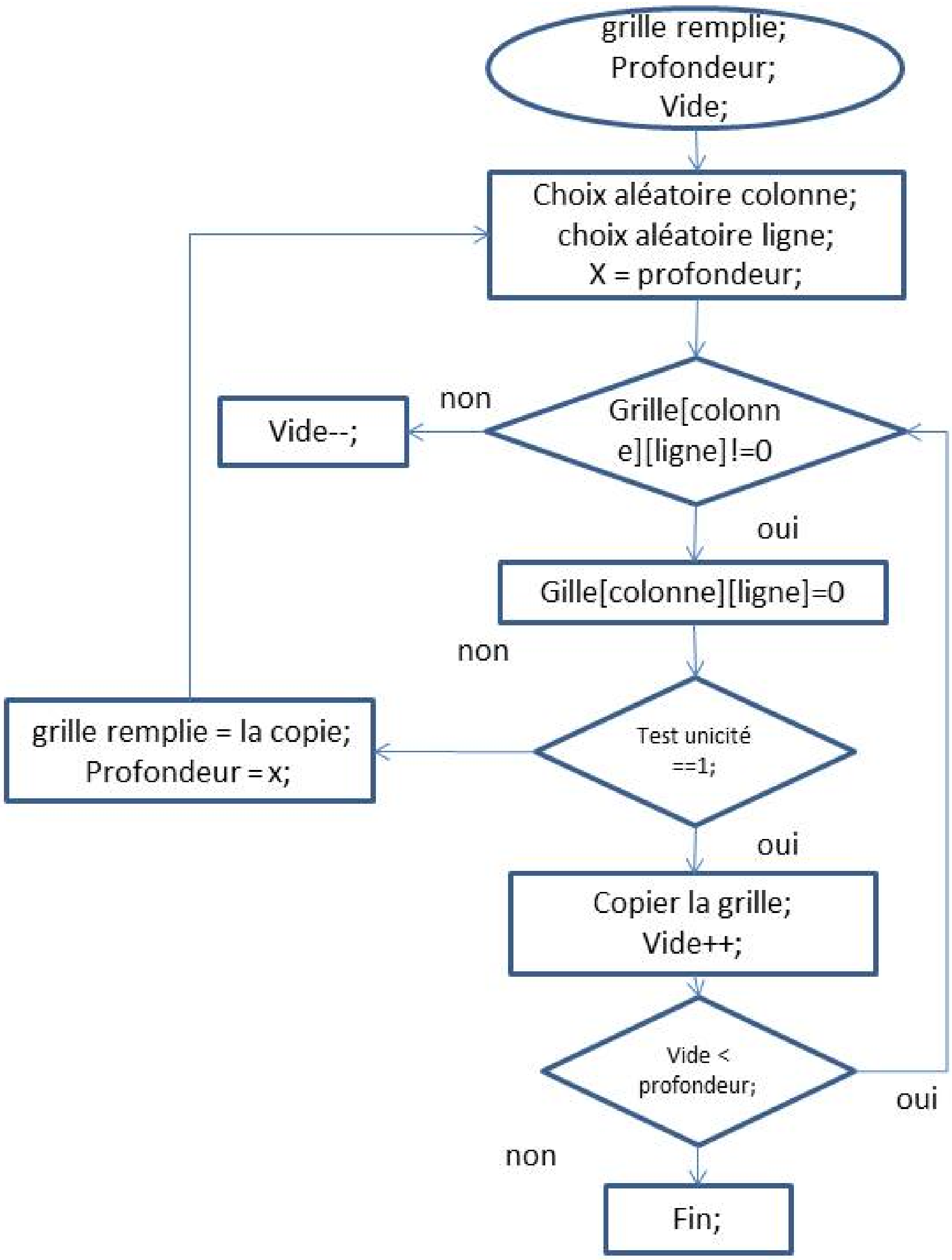
3. Il test toutes les combinaisons possibles pour la grille avec la case que l'on vient de vider, en faisant appel à une autre fonction qui vérifie combien on a de solution pour une grille donnée, alors deux cas sont possibles :

a) cas n°1 : Si la solution est unique, alors il continue l'algorithme en retournant au point 2, on vide une nouvelle case et on recommence jusqu'à ce qu'on ait le nombre de cases vides souhaité.

b) cas n°2 : S'il trouve plus qu'une seule solution, alors évidemment la solution n'est pas unique et donc il revient avant d'avoir vidé la case puis relance le point 2 en remettant le chiffre qui s'y trouvait puis recommence avec une nouvelle position de la case à vider.

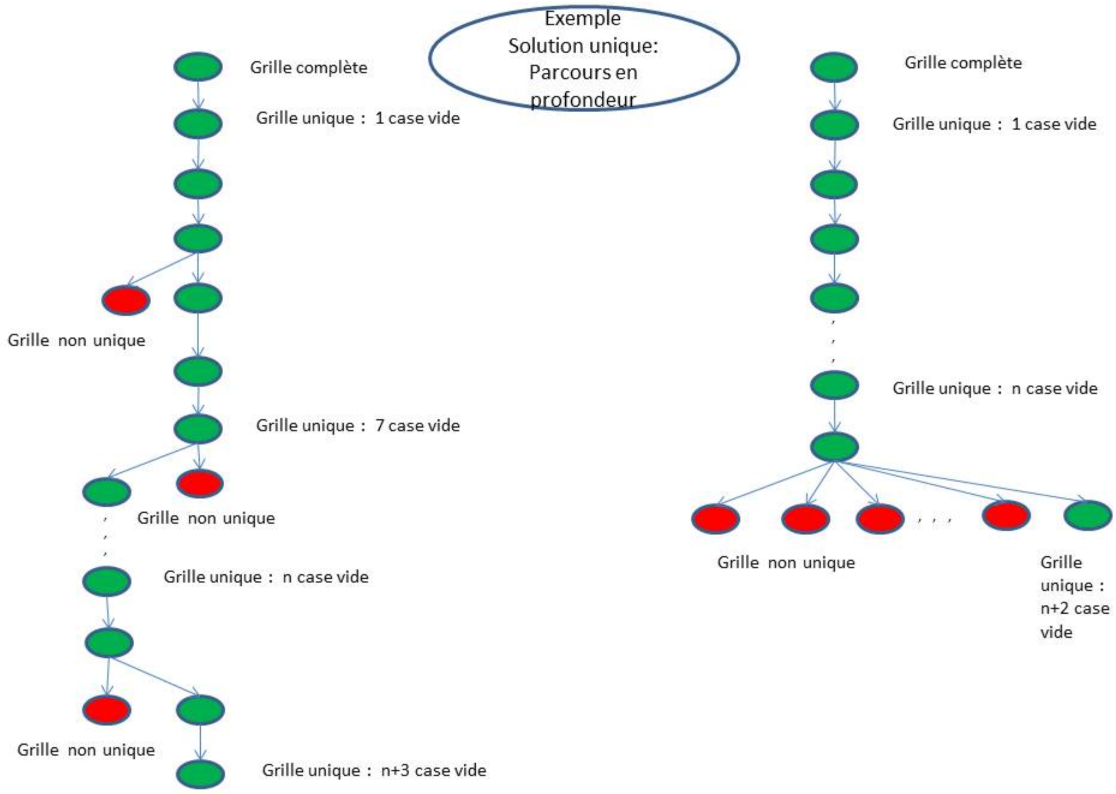
Le choix du nombre de case à vider dans la grille peut être modifié selon les paramètres de notre programme. selon Gary McGuire, il faut au moins 17 indices pour avoir l'unicité. Nous n'avons pas testé notre algorithme à cette profondeur à cause du temps d'exécution qui augmente à chaque case enlevée vu le processus aléatoire.

On peut représenter le fonctionnement de l'algorithme par un schéma :



3.1.1 Modélisation du problème avec des graphes.

Chaque point vert représente une grille avec une solution unique et chaque point rouge une grille avec plusieurs solutions.



3.2 La représentation mathématique d'une région.

La grille que nous allons voir dans cette partie nous permet de représenter l'appartenance d'une case à une région, c'est à dire une zone de la grille, par exemple un carré de 3x3 cases dans la version classique. La GrilleVoisin est une matrice de dimension $n^2 \times n^2$, avec n la dimension du Sudoku. Chaque ligne et colonne de GrilleVoisin représente une case du Sudoku.

GrilleVoisin est une matrice composée uniquement de 0 et de 1. Si une case de GrilleVoisin se situant à la ligne i et à la colonne j est égal à 1, alors le nombre se situant à la ligne a et sur la colonne b du Sudoku, avec a la partie entière de la division entre i et n, et b le reste de la division entre i et n, et le nombre se situant à la lignée c et sur la colonne d du Sudoku, avec c la partie entière de la division entre j et n et d le reste de la division entre j et n, ne doivent pas contenir la même valeur, elles sont considérées comme voisins.

Prenons un exemple, pour mieux comprendre, une grille classique 9×9. Chaque case de la matrice est numérotée comme ci-dessous :

0	1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16	17
18	19	20	21	22	23	24	25	26
27	28	29	30	31	32	33	34	35
36	37	38	39	40	41	42	43	44
45	46	47	48	49	50	51	52	53
54	55	56	57	58	59	60	61	62
63	64	65	66	67	68	69	70	71
72	73	74	75	76	77	78	79	80

Nous nous intéressons aux voisins de la toute première case de cette grille. Il faut donc que la valeur que contient cette case soit différente des valeurs présentes sur sa ligne, sa colonne et dans sa région, se sont les règles du Sudoku classique.

0	1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16	17
18	19	20	21	22	23	24	25	26
27	28	29	30	31	32	33	34	35
36	37	38	39	40	41	42	43	44
45	46	47	48	49	50	51	52	53
54	55	56	57	58	59	60	61	62
63	64	65	66	67	68	69	70	71
72	73	74	75	76	77	78	79	80

On peut remarquer ici que 0 n'est pas voisin de lui même, cela est important lors de la programmation.

La GrilleVoisin d'un Sudoku classique est donc un tableau de taille 81x81 qui contient pour la ligne 0 des 1 aux colonnes 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 18, 19, 20, 27, 36, 45, 54, 63 et 72. Le reste des cases de la ligne 0 sont nuls. Et ainsi de suite pour les 81 lignes du tableau, correspondant aux 81 cases de la grille de Sudoku.

La GrilleVoisin est générée qu'une seule fois et elle est générée rapidement. Une fois cette grille crée, on a à notre disposition tous les voisins des différentes cases. Grâce à la structure de GrilleVoisin, cela nous permet très simplement de créer de nombreuses variantes de Sudoku, il suffit de mettre des 1 ou des 0 là où il faut en fonction du format des régions du Sudoku correspondant.

4 Quelques variantes de Sudoku.

Dans cette partie nous allons décrire les différentes grilles de Sudoku que nous avons réalisé, nous nous sommes inspiré de documents[4] pour cela mais nous avons également crée des grilles originales. Nous expliquerons les différentes règles selon les grilles, et parlerons également de l'intérêt d'avoir créé ces variantes.

4.1 La grille classique.

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

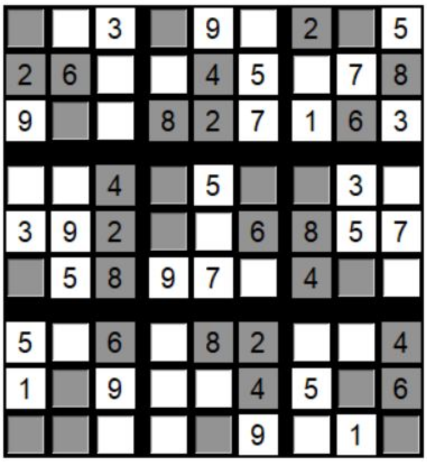
La grille classique est celle connue de tous, et contient 81 cases (9 lignes et 9 colonnes). Le but du jeu est de remplir ces cases avec des chiffres allant de 1 à 9 en veillant toujours à ce qu'un même chiffre ne figure qu'une seule fois par colonne, une seule fois par ligne, et une seule fois par région 3x3 de neuf cases.

4.2 La grille triple.

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

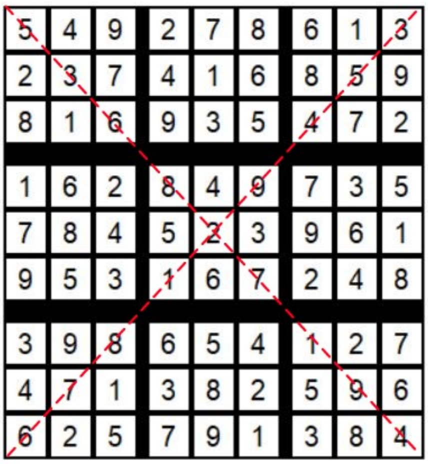
La grille triple est assez simple à comprendre et regroupe 3 grilles classiques. Ces trois grilles classiques partagent des régions de 9 cases, comme on peut le voir ci-dessus. Les règles restent les mêmes que pour une grille classique, mais on aura plus d'informations pour résoudre les régions partagées. Cette grille convient parfaitement aux personnes qui veulent toujours plus de sudoku.

4.3 Les grilles pairs/impairs



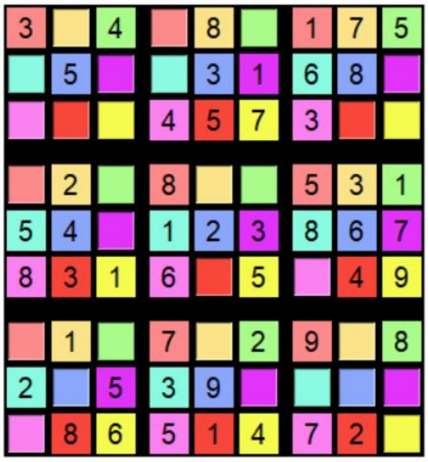
La grille paire/impaire possède les mêmes règles qu'une grille classique, mais une condition supplémentaire est imposée : les cases grisées doivent contenir un chiffre pair et les cases blanches un chiffre impair. Cela permet de donner une indication supplémentaire au joueur, et ainsi qu'il soit plus facile de la résoudre.

4.4 La grille diagonale.



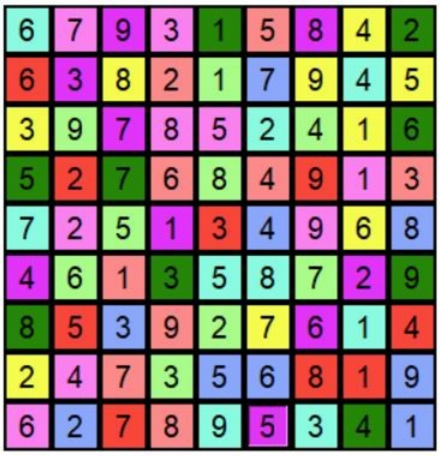
La grille diagonale respecte les mêmes règles qu'une grille classique, mais la condition supplémentaire ici est qu'un chiffre doit être présent une unique fois dans les diagonales dans lesquelles il est présent comme présenté ci-dessus.

4.5 La grille position.



La grille position suit également les mêmes règle qu'une grille classique, mais on ajoute à cela une nouvelle règle qui est qu'un même chiffre ne doit pas figurer en même position dans toutes les sous grilles 3x3. Nous avons ajouté des couleurs afin d'en faciliter la résolution. Cela revient à ce qu'un même chiffre doive figure une seule fois par colonne, une seule fois par ligne, une seule fois par carré de neuf cases et une seul fois par couleur, comme on peut le voir ci-dessus.

4.6 La grille couleur.



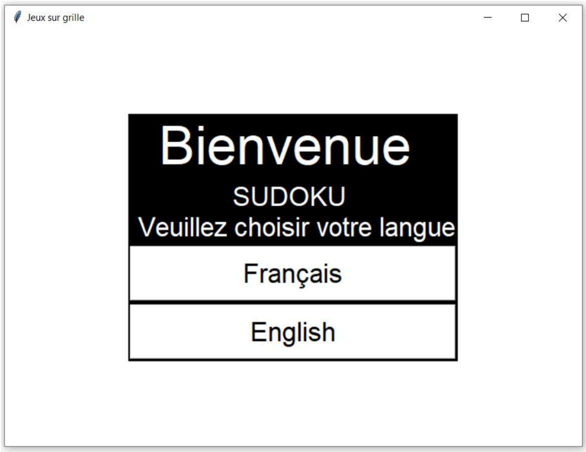
La grille couleur est une grille dans laquelle il faut remplir les cases avec des chiffres allant de 1 à 9 en veillant toujours à ce qu'un même chiffre ne figure qu'une seule fois par ligne, et une seule fois par couleur comme on peut le voir ci-dessus. Il n'y a pas de régions 3x3 comme dans les grilles classiques. Cette grille nous oblige à constamment chercher les couleurs pour pouvoir la résoudre. Étant plus compliquée, elle est idéale aux personnes ayant un bon niveau.

Nous avons créé de nombreuses variantes aux grilles de Sudoku classique. Parmi celles-ci se trouvent des versions plus ou moins difficiles à résoudre afin de convenir à tout type de joueur. Les joueurs débutants ayant du mal avec une grille classique peuvent maintenant jouer à des types de Sudoku plus simples telles que les grilles paire/impaire ou position. Pour les joueurs plus expérimentés, nous avons conçu des grilles plus difficiles à résoudre qui leur permettront de découvrir une nouvelle méthode de jeu, comme la grille couleur par exemple. Ces nouvelles variantes ont pour but d'actualiser le Sudoku classique et d'offrir aux joueurs des versions leur permettant de découvrir ou redécouvrir le jeu.

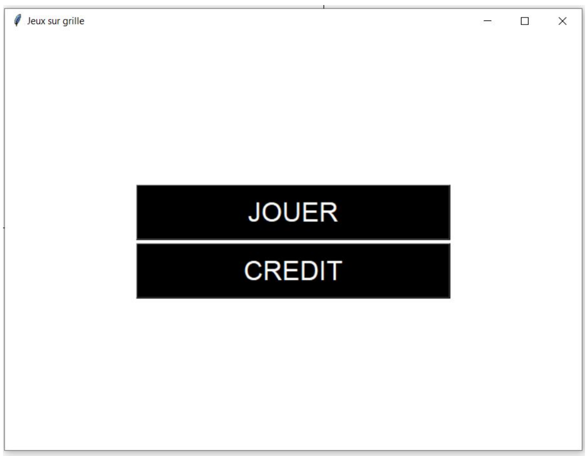
5 Présentation du logiciel

Dans cette partie nous allons découvrir ce que propose le logiciel.

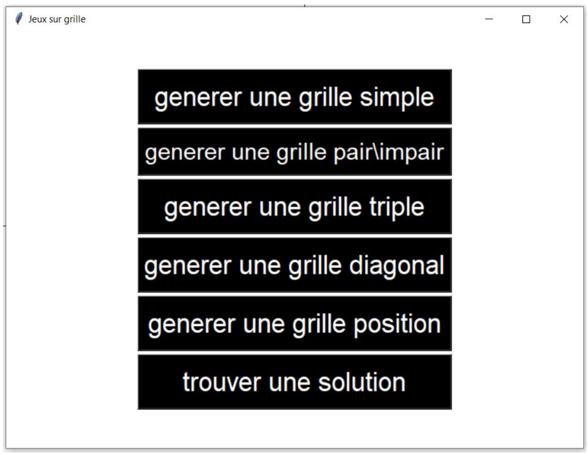
Dans un premier temps vous devez choisir dans quelle langue vous souhaitez continuer.



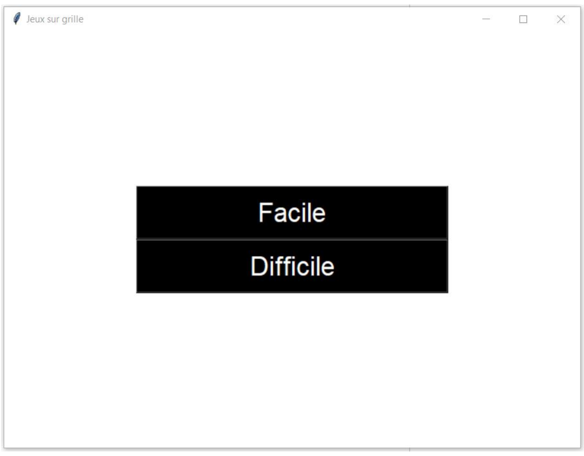
Ensuite s'affiche un menu qui vous laisse le choix entre commencer à jouer où lire les crédits. Si nous choisissons de lire les crédits on tombe sur un texte qui décrit comment s'est réalisé le projet.



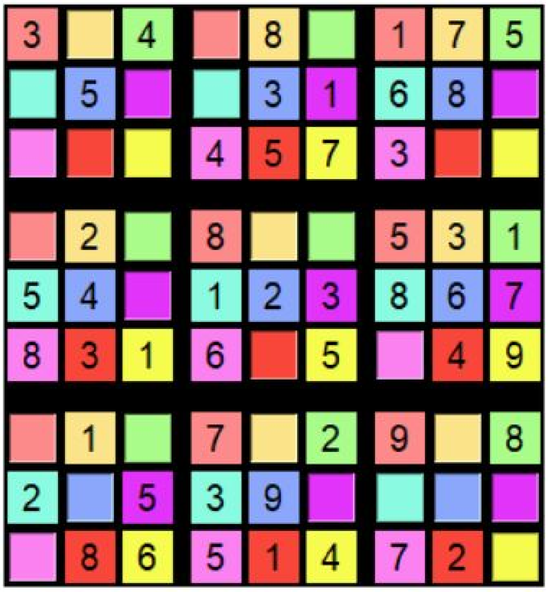
Si nous choisissons de jouer alors un nouveau menu se lance où vous devez sélectionner quelle type de Sudoku vous souhaitez afficher.



Certaines grilles vous propose par la suite un niveau de difficulté, en enlevant un certain nombre de cases :



Voilà par exemple ce qui s'affiche si vous avez choisi de générer une grille position :



6 Conclusion.

Même si au premier abord on pourrait croire que créer une grille de Sudoku revient juste à placer des chiffres en respectant quelques règles, nous nous sommes vite aperçu qu'il y avait une vraie méthode bien plus complexe, que les mathématiques derrière une grille de Sudoku sont tout autant voire plus techniques que ceux utilisés pour la résoudre. Dans le cas d'une publication dans un magazine ou une revue, il faut donc que la solution soit unique, mais aussi que la grille soit solvable et cela demande de comprendre comment utiliser des méthodes à base de backtracking, qui joue un rôle très important dans la génération des grilles de Sudoku.

Pour ce qui est de la programmation nous nous sommes réparti le travail en plusieurs sous-groupes avec un connecteur qui avait une vision d'ensemble et essayait de dépanner lorsque nous étions bloqués mais personne n'étant vraiment à l'aise en python nous avons plusieurs fois bloqué pour développer certaines fonctions, par exemple les tests d'unicité ou l'interface graphique.

Une fois le plan du rapport dressé nous avons réparti le travail entre tous les membres du projet en essayant d'anticiper les parties que chacun allait présenter à l'oral mais même en nous réunissant régulièrement nous n'avons à l'heure actuelle pas encore décidé de qui présente quoi à l'oral.

Il existe un grand nombre de variantes que nous aurions pu implémenter avec plus de temps, ou encore améliorer l'interface graphique et proposer d'autres fonctionnalitées, aussi la programmation modulaire que nous avons choisi nous permet si l'on veut de proposer par exemple des grilles en ligne avec vérification des chiffres rentrés sans avoir à repartir de zéro.

Bonus : une grille un peu atypique spéciale π day.

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

Références

1 Jean-Paul Delahaye. Pour la science-logique et calcul-le tsunami du sudoku. 2005.

2 Emerging Technology from the arXivarchive page. Mathematicians solve minimum sudoku problem. 2012.

3 Gary McGuire. Computer science-data structures and algorithms. 2005.

4 Numberphile. 17 and Sudoku Clues - Numberphile. 2012.