

Sudoku et

Logique

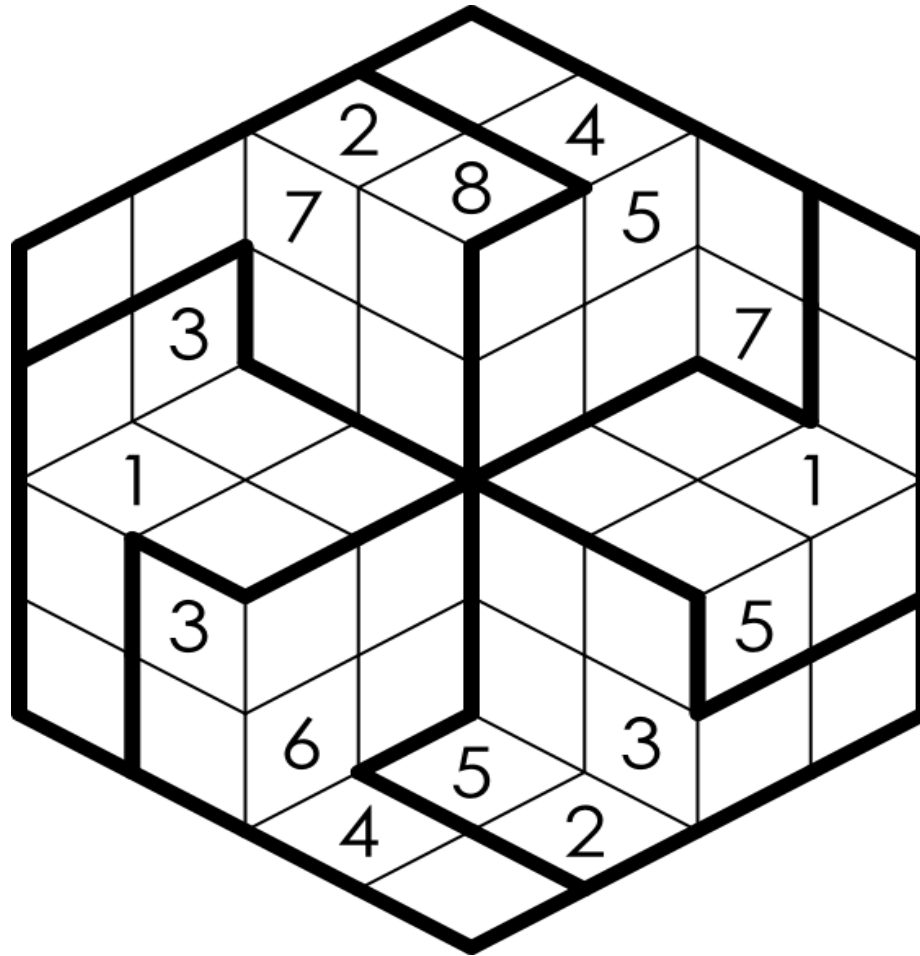
Propositionnelle :

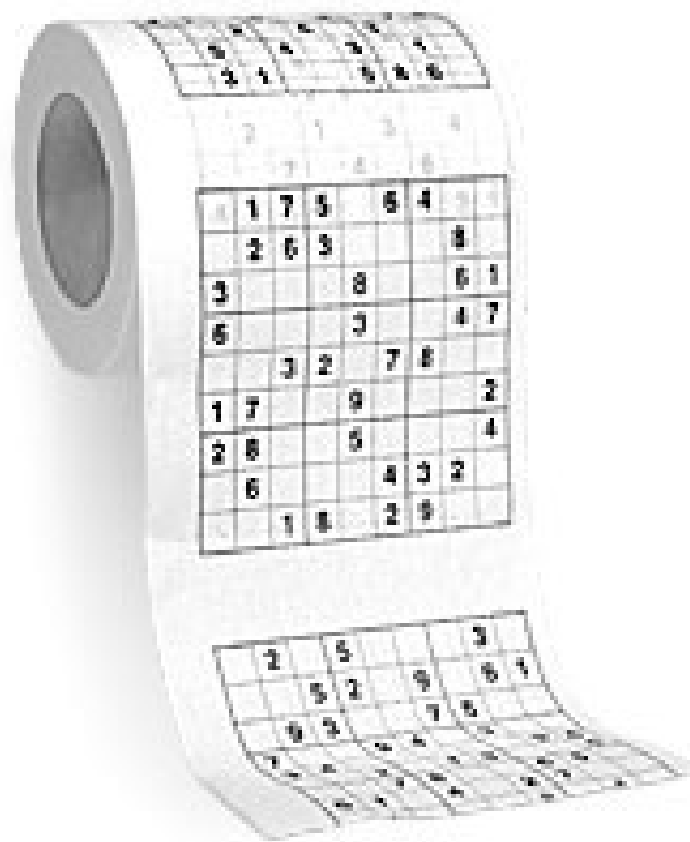
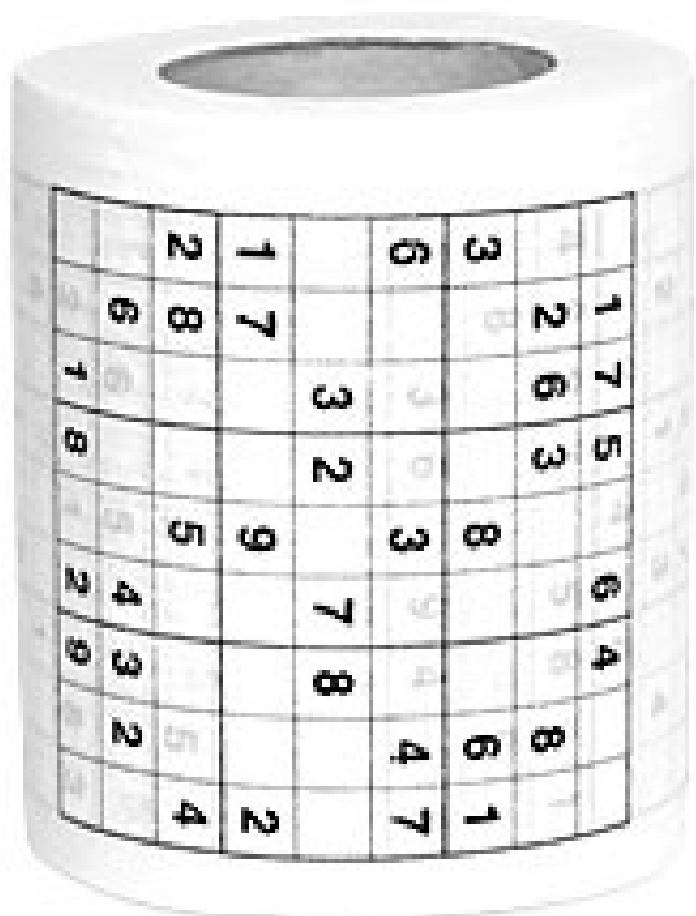
Les règles du sudoku

Sudoku

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

Exemples de sudokus





Algobox

```
Code de l'algorithme

VARIABLES
a EST_DU_TYPE NOMBRE
b EST_DU_TYPE NOMBRE
c EST_DU_TYPE NOMBRE
d EST_DU_TYPE NOMBRE
e EST_DU_TYPE NOMBRE
f EST_DU_TYPE NOMBRE
g EST_DU_TYPE NOMBRE
h EST_DU_TYPE NOMBRE
i EST_DU_TYPE NOMBRE
j EST_DU_TYPE NOMBRE
k EST_DU_TYPE NOMBRE
l EST_DU_TYPE NOMBRE
m EST_DU_TYPE NOMBRE
n EST_DU_TYPE NOMBRE
o EST_DU_TYPE NOMBRE
p EST_DU_TYPE NOMBRE
ligne_1 EST_DU_TYPE NOMBRE
grille EST_DU_TYPE NOMBRE
ligne_2 EST_DU_TYPE NOMBRE
ligne_3 EST_DU_TYPE NOMBRE
ligne_4 EST_DU_TYPE NOMBRE

DEBUT_ALGORITHME
LIRE a
LIRE b
LIRE c
LIRE d
LIRE e
LIRE f
LIRE g
LIRE h
LIRE i
LIRE j
LIRE k
LIRE l
LIRE m
LIRE n
LIRE o
LIRE p
```

Nous avons entré les variables

Puis entré les contraintes en langage informatique

```
Code de l'algorithme

SI (0==0) ALORS
  DEBUT_SI
  SI (a!=b ET a!=c ET a!=d ET b!=c ET c!=d ET b!=d) ALORS
    DEBUT_SI
    AFFICHER "ligne_1 valide"
    FIN_SI
  SINON
    DEBUT_SINON
    AFFICHER "grille impossible"
    FIN_SINON
  SI (e!=f ET e!=g ET e!=h ET f!=g ET f!=h ET g!=h) ALORS
    DEBUT_SI
    AFFICHER "ligne_2 valide"
    FIN_SI
  SINON
    DEBUT_SINON
    AFFICHER "grille impossible"
    FIN_SINON
  SI (i!=j ET i!=k ET i!=l ET j!=k ET j!=l ET k!=l) ALORS
    DEBUT_SI
    AFFICHER "ligne_3 valide"
    FIN_SI
  SINON
    DEBUT_SINON
    AFFICHER "grille impossible"
    FIN_SINON
  SI (m!=n ET m!=o ET m!=p ET n!=o ET n!=p ET o!=p) ALORS
    DEBUT_SI
    AFFICHER "ligne_4 valide"
    FIN_SI
  SINON
    DEBUT_SINON
    AFFICHER "grille impossible"
    FIN_SINON
  FIN_SI
SINON
  DEBUT_SINON
  AFFICHER "La grille est impossible"
  FIN_SINON
```

Algorithme pour les lignes

Code de l'algorithme

VARIABLES

- B EST_DU_TYPE NOMBRE
- E EST_DU_TYPE NOMBRE
- F EST_DU_TYPE NOMBRE
- S EST_DU_TYPE NOMBRE
- A EST_DU_TYPE NOMBRE
- V EST_DU_TYPE NOMBRE

DEBUT_ALGORITHME

- LIRE A
- LIRE B
- LIRE E
- LIRE F
- S PREND_LA_VALEUR $A+B+E+F$

SI (A==B OU E==F) ALORS

- DEBUT_SI
- AFFICHER "FAUX"
- FIN_SI

SINON

- DEBUT_SINON

SI (S>10 OU S<10) ALORS

- DEBUT_SI
- AFFICHER "Faux"
- FIN_SI

SINON

- DEBUT_SINON
- AFFICHER "Vrai"
- FIN_SINON

- FIN_SINON

FIN_ALGORITHME

Comment savoir si une grille est correcte ?

•Le langage informatique de Solveur SAT

Ces images sont des contraintes extraites du **logiciel Solveur SAT**

$$\bigwedge_{i \in \{1, \dots, 9\}} \bigwedge_{j \in \{1, \dots, 9\}} \bigvee_{n \in \{1, \dots, 9\}} (p_{ij n})$$

-Les chiffres du tableau doivent être des entiers compris entre 1 et 9

$$\bigwedge_{n \in \{1, \dots, 9\}} \bigvee_{i \in \{7, \dots, 9\}} \bigvee_{j \in \{7, \dots, 9\}} (p_{ij n})$$

-Dans le bloc en bas à droite, tous les chiffres (de 1 à 9) doivent être présents

La grille de sudoku diabolique à résoudre

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

Représentation de la case d'une grille de sudoku grâce au langage informatique de Solveur SAT du type (p i j n)

(p 6 7 9)

(p 9 5 4)

• Résultat du sudoku diabolique

Cela correspond au numéro de la ligne de la grille

La formule est satisfiable, voici une solution :

<i>P</i> _{1,1,1}	<i>P</i> _{1,2,9}	<i>P</i> _{1,3,7}	<i>P</i> _{1,4,6}	<i>P</i> _{1,5,5}	<i>P</i> _{1,6,3}	<i>P</i> _{1,7,8}	
<i>P</i> _{1,8,2}	<i>P</i> _{1,9,4}	<i>P</i> _{2,1,2}	<i>P</i> _{2,2,6}	<i>P</i> _{2,3,3}	<i>P</i> _{2,4,4}	<i>P</i> _{2,5,8}	<i>P</i> _{2,6,1}
<i>P</i> _{2,7,7}	<i>P</i> _{2,8,5}	<i>P</i> _{2,9,9}	<i>P</i> _{3,1,8}	<i>P</i> _{3,2,4}	<i>P</i> _{3,3,5}	<i>P</i> _{3,4,9}	<i>P</i> _{3,5,2}
<i>P</i> _{3,6,7}	<i>P</i> _{3,7,6}	<i>P</i> _{3,8,3}	<i>P</i> _{3,9,1}	<i>P</i> _{4,1,5}	<i>P</i> _{4,2,2}	<i>P</i> _{4,3,4}	<i>P</i> _{4,4,8}
<i>P</i> _{4,5,1}	<i>P</i> _{4,6,9}	<i>P</i> _{4,7,3}	<i>P</i> _{4,8,7}	<i>P</i> _{4,9,6}	<i>P</i> _{5,1,7}	<i>P</i> _{5,2,8}	<i>P</i> _{5,3,9}
<i>P</i> _{5,4,5}	<i>P</i> _{5,5,3}	<i>P</i> _{5,6,6}	<i>P</i> _{5,7,4}	<i>P</i> _{5,8,1}	<i>P</i> _{5,9,2}	<i>P</i> _{6,1,3}	<i>P</i> _{6,2,1}
<i>P</i> _{6,3,6}	<i>P</i> _{6,4,2}	<i>P</i> _{6,5,7}	<i>P</i> _{6,6,4}	<i>P</i> _{6,7,9}	<i>P</i> _{6,8,8}	<i>P</i> _{6,9,5}	<i>P</i> _{7,1,9}
<i>P</i> _{7,2,7}	<i>P</i> _{7,3,1}	<i>P</i> _{7,4,3}	<i>P</i> _{7,5,6}	<i>P</i> _{7,6,2}	<i>P</i> _{7,7,5}	<i>P</i> _{7,8,4}	<i>P</i> _{7,9,8}
<i>P</i> _{8,1,4}	<i>P</i> _{8,2,5}	<i>P</i> _{8,3,2}	<i>P</i> _{8,4,7}	<i>P</i> _{8,5,9}	<i>P</i> _{8,6,8}	<i>P</i> _{8,7,1}	<i>P</i> _{8,8,6}
<i>P</i> _{8,9,3}	<i>P</i> _{9,1,6}	<i>P</i> _{9,2,3}	<i>P</i> _{9,3,8}	<i>P</i> _{9,4,1}	<i>P</i> _{9,5,4}	<i>P</i> _{9,6,5}	<i>P</i> _{9,7,2}
<i>P</i> _{9,8,9}	<i>P</i> _{9,9,7}						

Cela correspond au chiffre présent dans la case de la grille

Cela correspond au numéro de la colonne de la grille

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