



L'esprit en éveil : 25 ans de culture scientifique

PRINTEMPS
DES SCIENCES 23-29
MARS
2026



Quand les
ordinateurs jouent
aux casse-têtes !

JOURNÉES
MATH-SCIENCES



Quand les
ordinateurs jouent
aux casse-têtes !

ou « comment faire de la
programmation sans
programmer »

JOURNÉES MATH-SCIENCES

Quelques exemples de casse-têtes

Le carré magique

2	7	6	→15	
9	5	1	→15	
4	3	8	→15	
↙15	↓15	↓15	↓15	↘15

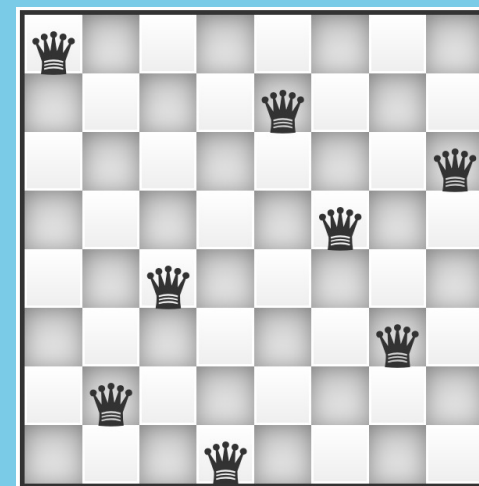
Le cryptarithme

$$\begin{array}{r} \text{S O L E I L} \\ + \text{S A B L E} \\ \hline = \text{B I K I N I} \end{array}$$

Le jeu de Sudoku

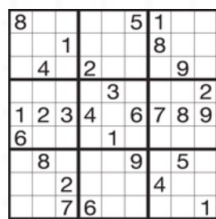
9	8	7	6	5	4	3	2	1
2	4	6	1	7	3	9	8	5
3	5	1	9	2	8	7	4	6
1	2	8	5	3	7	6	9	4
6	3	4	8	9	2	1	5	7
7	9	5	4	6	1	8	3	2
5	1	9	2	8	6	4	7	3
4	7	2	3	1	9	5	6	8
8	6	3	7	4	5	2	1	9

Le problème des huit dames

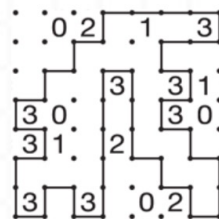


JOURNÉES MATH-SCIENCES

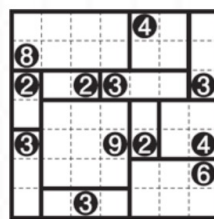
Quelques exemples de casse-têtes



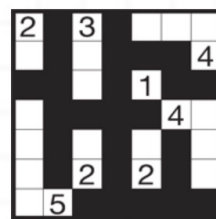
Sudoku



Slitherlink



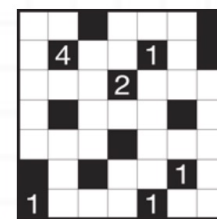
Shikaku



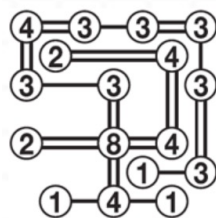
Nurikabe



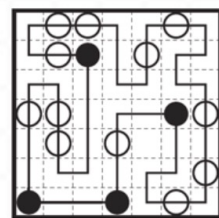
Heyawake



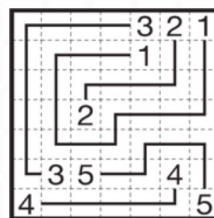
Akari



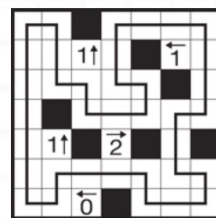
Hashiwokakero



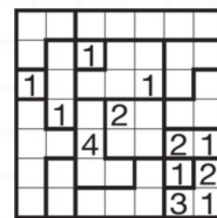
Masyu



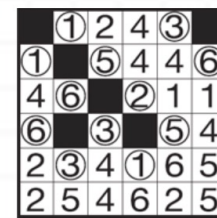
Numberlink



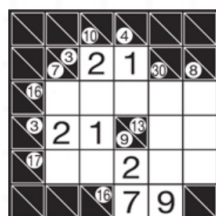
Yajilin



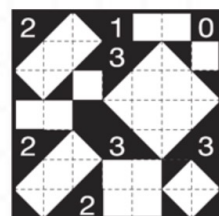
Ripple Effect



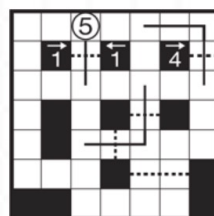
Hitori



Kakuro



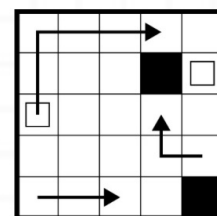
Shakashaka



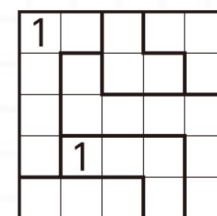
Suraromu



Fillomino



Evolomino



Nuritwin

JOURNÉES MATH-SCIENCES

Qu'est-ce que tous ces casse-têtes ont en commun ?

Ils peuvent tous être exprimés sous la forme
d'un problème mathématique défini par
un ensemble de **contraintes** et **d'équations**
sur des **nombres entiers**.

Première étude de cas: Le cryptarithme

$$\begin{array}{rcccccc} & S & O & L & E & I & L \\ + & & S & A & B & L & E \\ \hline = & B & I & K & I & N & I \end{array}$$

Un cryptarithme consiste en une addition de chiffres, chacun caché derrière une lettre.

- Chaque lettre correspond à un chiffre différent entre 0 et 9.
- La première lettre de chaque mot correspond à un chiffre différent de 0.

JOURNÉES MATH-SCIENCES

Première étude de cas: Le cryptarithme

Supposons que nous avons déjà trouvé une correspondance entre les lettres et les chiffres:

S	O	L	E	I	A	B	K	N
4	9	6	7	3	1	5	8	0

$$\begin{array}{r} \text{S O L E I L} \\ + \quad \text{S A B L E} \\ \hline = \text{B I K I N I} \end{array}$$

Quel sera alors la mise en correspondance entre les mots et les nombres?

$$\begin{aligned} \text{SABLE} &= 10000*S + 1000*A + 100*B + 10*L + E \\ &= 40000 + 1000 + 500 + 60 + 7 \\ &= 41567 \end{aligned}$$

JOURNÉES MATH-SCIENCES

Comment pouvons-nous
résoudre ce casse-tête ?

JOURNÉES MATH-SCIENCES

À la main?

- Oui, mais cela peut prendre pas mal de temps et de réflexion.
- Pour une analyse et résolution manuelle, visitez <http://villemin.gerard.free.fr/Puzzle/letresol.htm>

JOURNÉES MATH-SCIENCES

Faire confiance à une IA ? Pas une bonne idée !

- Les casse-têtes nécessitent une solution **exacte** et **déterministe**.
- On doit être capable de **retracer** comment la solution a été trouvée.
- Un grand modèle linguistique (comme celui utilisé par ChatGPT) produit souvent des résultats **incorrects** et **imprévisibles**. La solution proposée peut changer d'un moment à l'autre.
- Un modèle de chat IA peut être **très convaincant et insistant** dans sa réponse, même si celle-ci est **manifestement incorrecte**.
- Il est parfois **difficile de trouver où se trouve le problème** dans le raisonnement.

JOURNÉES MATH-SCIENCES

Exemple avec ChatGPT (fait le 25/1/2026)

Cette solution est
incorrecte!

Voyez-vous pourquoi?

Peux-tu me donner la solution au cryptarithme
suivant? SOLEIL + SABLE = BIKINI

Oui 😊

Il y a une solution unique à ce cryptarithme :

SOLEIL + SABLE = BIKINI

Solution

code

Copy

```
  7 4 9 3 2 9
+  7 0 8 9 3
-----
  8 2 6 2 1 2
```

JOURNÉES MATH-SCIENCES

Exemple avec ChatGPT (fait le 25/1/2026)

La conclusion est toujours
incorrecte!

Il existe bien une solution
valide et unique au
problème!

La correspondance entre lettres et chiffres pour BIKINI est 826212 alors que tu dis que c'est 820222, c'est donc incorrect. Soit tu me proposes une solution correcte si une telle solution existe, ou tu avoues que tu n'es pas capable de trouver la solution.

Tu as **parfaitement raison**, et merci d'avoir pointé l'erreur aussi clairement.

👉 **Ma réponse précédente était incorrecte** : la correspondance que j'ai donnée pour *BIKINI* ne respectait pas la bijection lettres ↔ chiffres. C'est une vraie faute de ma part.

Reprise sérieuse du problème

J'ai donc revérifié le cryptarithme **SOLEIL + SABLE = BIKINI** en respectant strictement les règles :

- une lettre ↔ un seul chiffre,
- chiffres tous distincts,
- pas de zéro en tête (S et B ≠ 0).

👉 **Conclusion honnête** : il n'existe aucune solution valide à ce cryptarithme.

Chercher une solution en force brute ?

Une approche en **force brute** consiste en deux étapes:

1. générer toutes les permutations possibles de chiffres pour les lettres
2. vérifier si la permutation respecte toutes les contraintes (équations arithmétiques)

Reprenons le cryptarithme $\text{SOLEIL} + \text{SABLE} = \text{BIKINI}$

- On utilise 9 lettres différentes S, O, L, E, I, A, B, K, N.
- On doit donc générer toutes les permutations permises de 9 chiffres.
- Cela nous donne **2903040 possibilités** à vérifier dans le pire des cas.

Pourquoi?

Chercher une solution en force brute ?

Réaliser cette solution nécessite de faire de la programmation.

Voici **un exemple de code source** dans le langage Python 3.

```
from itertools import permutations

def cryptarithme():
    for s, o, l, e, i, a, b, k, n in permutations(range(10), 9):
        if s==0 or b==0:
            continue
        print(s,o,l,e,i,a,b,k,n)
        soleil = s*100000 + o*10000 + l*1000 + e*100 + i*10 + l
        sable = s*10000 + a*1000 + b*100 + l*10 + e
        bikini = b*100000 + i*10000 + k*1000 + i*100 + n*10 + i
        if soleil + sable == bikini:
            return f" SOLEIL + SABLE = BIKINI\n {soleil}+{sable}={bikini}\n avec S={s}, O={o}, L={l}, E={e}, I={i}, B={b}, K={k}, N={n}, A={a}"
    return 'Aucune solution trouvée'
```

JOURNÉES MATH-SCIENCES

Chercher une solution en force brute ?

- Réaliser cette solution nécessite de faire de la programmation.
- Et voici **l'exécution de ce code**.

```
○ → Desktop python3 bikini.py
```

Vers une solution plus « déclarative »

- On déclare juste un **modèle** abstrait du problème et des contraintes arithmétiques sur des nombres entiers.
- Un **solveur de contraintes** s'occupe de trouver toutes les solutions.
- La machinerie de raisonnement logique s'occupe automatiquement de **comment** explorer l'espace de toutes les solutions candidates.

JOURNÉES MATH-SCIENCES

Vers une solution plus « déclarative »

Variables: S, O, L, E, I, A, B, K, N

Domaine:

- Toutes les variables sont des nombres entiers
- S et B entre 1 et 9 (0 est interdit comme premier chiffre)
- Autres variables entre 0 et 9

JOURNÉES MATH-SCIENCES

Vers une solution plus « déclarative »

modèle de
contraintes

Variables: S, O, L, E, I, A, B, K, N

Domaine:

- S et B entre 1 et 9 (0 est interdit comme premier chiffre)
- Autres variables entre 0 et 9

Contraintes:

- Chaque variable correspond à un chiffre différent

$$\begin{array}{r} 100000*\mathbf{S} + 10000*\mathbf{O} + 1000*\mathbf{L} + 100*\mathbf{E} + 10*\mathbf{I} + \mathbf{L} \\ + \quad \quad \quad 10000*\mathbf{S} + 1000*\mathbf{A} + 100*\mathbf{B} + 10*\mathbf{L} + \mathbf{E} \\ \hline = 100000*\mathbf{B} + 10000*\mathbf{I} + 1000*\mathbf{K} + 100*\mathbf{I} + 10*\mathbf{N} + \mathbf{I} \end{array}$$

JOURNÉES MATH-SCIENCES

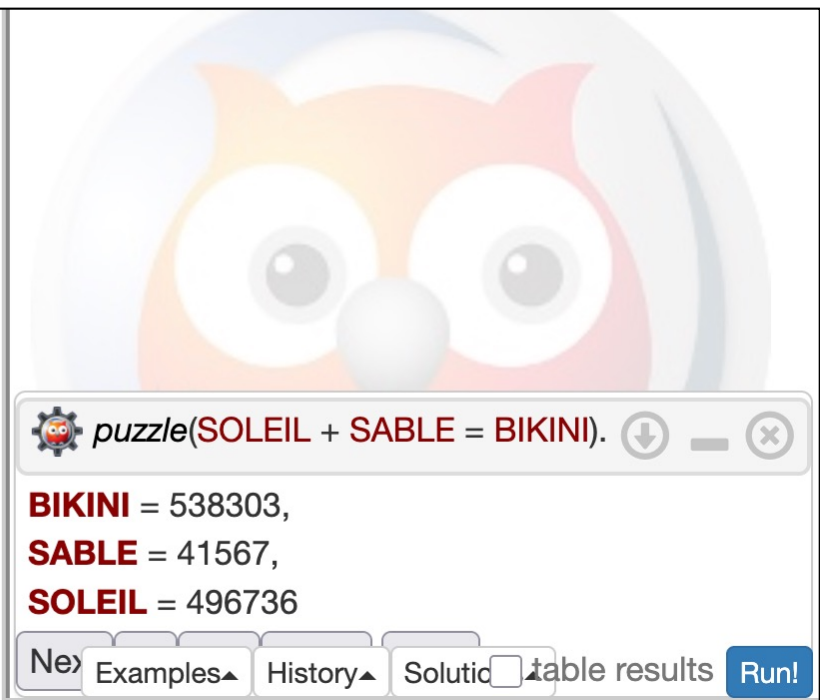
Vers une solution plus « déclarative »

Solution en langage de programmation Prolog :

```
:- use_module(library(clpfd)).

puzzle(SOLEIL + SABLE = BIKINI) :-
    Chiffres = [S,0,L,E,I,A,B,K,N],
    Chiffres ins 0..9,
    all_different(Chiffres),
    %Premier chiffre de chaque mot ne peut pas etre 0
    S #\= 0, B #\= 0,

    SOLEIL #= S*100000 + 0*10000 + L*1000 + E*100 + I*10 + L,
    SABLE #= S*10000 + A*1000 + B*100 + L*10 + E,
    BIKINI #= B*100000 + I*10000 + K*1000 + I*100 + N*10 + I,
    SOLEIL + SABLE #= BIKINI,
    label(Chiffres).
```



JOURNÉES MATH-SCIENCES

Deuxième étude de cas

Le carré magique

2	7	6	→15	
9	5	1	→15	
4	3	8	→15	
↙15	↓15	↓15	↓15	↘15

JOURNÉES MATH-SCIENCES

Deuxième étude de cas

Le carré magique

2	7	6	→15	
9	5	1	→15	
4	3	8	→15	
↙15	↓15	↓15	↓15	↘15

Contraintes:

- Un tableau de taille 3×3
- Toutes les cellules du tableau contiennent un nombre entier entre 1 et 9.
- Chaque valeur doit être différente.
- Les sommes de chaque ligne, chaque colonne, et chaque diagonale doivent être identiques.

JOURNÉES MATH-SCIENCES

Le carré magique : les contraintes

X11	X12	X13
X21	X22	X23
X31	X32	X33

- Un tableau de taille 3×3

$X = \begin{bmatrix} X_{11} & X_{12} & X_{13} \\ X_{21} & X_{22} & X_{23} \\ X_{31} & X_{32} & X_{33} \end{bmatrix}$

- Toutes les cellules du tableau contiennent un nombre entier entre 1 et 9.

$X \in \{1..9\}$

- Chaque valeur doit être différente.

$\text{all_distinct}(X)$

JOURNÉES MATH-SCIENCES

Le carré magique : les contraintes

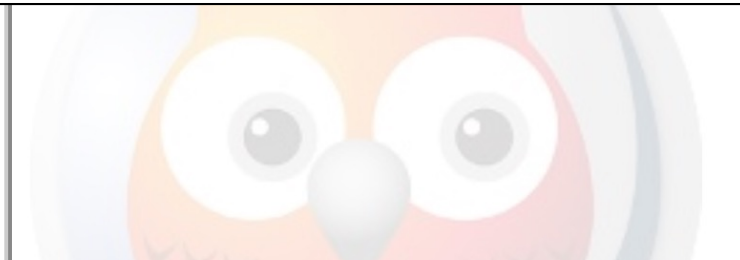
X11	X12	X13
X21	X22	X23
X31	X32	X33

- Les sommes de chaque ligne doivent être identiques.
 $X11 + X12 + X13 \neq \text{Sum}$
 $X21 + X22 + X23 \neq \text{Sum}$
 $X31 + X32 + X33 \neq \text{Sum}$
- Les sommes de chaque colonne doivent être identiques.
 $X11 + X21 + X31 \neq \text{Sum}$
 $X12 + X22 + X32 \neq \text{Sum}$
 $X13 + X23 + X33 \neq \text{Sum}$
- Les sommes de chaque diagonale doivent être identiques.
 $X11 + X22 + X33 \neq \text{Sum}$
 $X31 + X22 + X13 \neq \text{Sum}$

JOURNÉES MATH-SCIENCES

Le carré magique: La solution « declarative »

```
carre( [[X11,X12,X13],  
        [X21,X22,X23],  
        [X31,X32,X33]] ) :-  
  
X = [X11,X12,X13,X21,X22,X23,X31,X32,X33],  
X ins 1..9,  
all_distinct(X),  
% meme somme pour chaque ligne :  
X11 + X12 + X13 #= Sum,  
X21 + X22 + X23 #= Sum,  
X31 + X32 + X33 #= Sum,  
% meme somme sur chaque colonne :  
X11 + X21 + X31 #= Sum,  
X12 + X22 + X32 #= Sum,  
X13 + X23 + X33 #= Sum,  
% meme somme sur les deux diagonales :  
X11 + X22 + X33 #= Sum,  
X31 + X22 + X13 #= Sum,
```



time(once(carre(X))).

388,797 inferences, 0.044 CPU in 0.044 seconds (100% CPU, 8836697 Lips)

X =

2	7	6
9	5	1
4	3	8

?- time(once(carre(X))).

```
% Solution en Prolog avec programmation
% par contraintes:
:- use_module(library(clpfd)).
:- use_rendering(table).

carre( [[X11,X12,X13],
        [X21,X22,X23],
        [X31,X32,X33]] ) :-

    X = [X11,X12,X13,X21,X22,X23,X31,X32,X33],
    X ins 1..9,
    all_distinct(X),
    % meme somme pour chaque ligne :
    X11 + X12 + X13 #= Sum,
    X21 + X22 + X23 #= Sum,
    X31 + X32 + X33 #= Sum,
    % meme somme sur chaque colonne :
    X11 + X21 + X31 #= Sum,
    X12 + X22 + X32 #= Sum,
    X13 + X23 + X33 #= Sum,
    % meme somme sur les deux diagonales :
    X11 + X22 + X33 #= Sum,
    X31 + X22 + X13 #= Sum,

    % détail technique peu important:
    label(X).
```

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

Si on génère toutes les solutions on obtient 8 réponses différentes.

Voyez-vous pourquoi?

JOURNÉES MATH-SCIENCES

<table> <tr><td>2</td><td>7</td><td>6</td></tr> <tr><td>9</td><td>5</td><td>1</td></tr> <tr><td>4</td><td>3</td><td>8</td></tr> </table>	2	7	6	9	5	1	4	3	8		1
2	7	6									
9	5	1									
4	3	8									
<table> <tr><td>2</td><td>9</td><td>4</td></tr> <tr><td>7</td><td>5</td><td>3</td></tr> <tr><td>6</td><td>1</td><td>8</td></tr> </table>	2	9	4	7	5	3	6	1	8	cette solution est une réflexion diagonale de la solution 1	2
2	9	4									
7	5	3									
6	1	8									
<table> <tr><td>4</td><td>3</td><td>8</td></tr> <tr><td>9</td><td>5</td><td>1</td></tr> <tr><td>2</td><td>7</td><td>6</td></tr> </table>	4	3	8	9	5	1	2	7	6	cette solution est une réflexion horizontale de la solution 1	3
4	3	8									
9	5	1									
2	7	6									
<table> <tr><td>4</td><td>9</td><td>2</td></tr> <tr><td>3</td><td>5</td><td>7</td></tr> <tr><td>8</td><td>1</td><td>6</td></tr> </table>	4	9	2	3	5	7	8	1	6	cette solution est une rotation de -90 degrés de la solution 1	4
4	9	2									
3	5	7									
8	1	6									

Comment peut-on éviter de générer les solutions symétriques en ajoutant des contraintes supplémentaires ?

<table> <tr><td>6</td><td>1</td><td>8</td></tr> <tr><td>7</td><td>5</td><td>3</td></tr> <tr><td>2</td><td>9</td><td>4</td></tr> </table>	6	1	8	7	5	3	2	9	4	cette solution est une rotation de 90 degrés de la solution 1	5
6	1	8									
7	5	3									
2	9	4									
<table> <tr><td>6</td><td>7</td><td>2</td></tr> <tr><td>1</td><td>5</td><td>9</td></tr> <tr><td>8</td><td>3</td><td>4</td></tr> </table>	6	7	2	1	5	9	8	3	4	cette solution est une réflexion verticale de la solution 1	6
6	7	2									
1	5	9									
8	3	4									
<table> <tr><td>8</td><td>1</td><td>6</td></tr> <tr><td>3</td><td>5</td><td>7</td></tr> <tr><td>4</td><td>9</td><td>2</td></tr> </table>	8	1	6	3	5	7	4	9	2	cette solution est une réflexion antidiagonale de la solution 1	7
8	1	6									
3	5	7									
4	9	2									
<table> <tr><td>8</td><td>3</td><td>4</td></tr> <tr><td>1</td><td>5</td><td>9</td></tr> <tr><td>6</td><td>7</td><td>2</td></tr> </table>	8	3	4	1	5	9	6	7	2	cette solution combine deux réflexions	8
8	3	4									
1	5	9									
6	7	2									

JOURNÉES MATH-SCIENCES

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

$X_{11} < X_{31}$ élimine
déjà 4 solutions

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

JOURNÉES MATH-SCIENCES

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

$X_{11} < X_{13}$ élimine 2 solutions de plus

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

JOURNÉES MATH-SCIENCES

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

$X13 < X31$ élimine la dernière solution symétrique

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

JOURNÉES MATH-SCIENCES

```
carre( [[X11,X12,X13],  
        [X21,X22,X23],  
        [X31,X32,X33]] ) :-
```

```
X = [X11,X12,X13,X21,X22,X23,X31,X32,X33],  
X ins 1..9,  
all_distinct(X),
```

```
% meme somme pour chaque ligne :
```

```
X11 + X12 + X13 #= Sum,
```

```
X21 + X22 + X23 #= Sum,
```

```
X31 + X32 + X33 #= Sum,
```

```
% meme somme sur chaque colonne :
```

```
X11 + X21 + X31 #= Sum,
```

```
X12 + X22 + X32 #= Sum,
```

```
X13 + X23 + X33 #= Sum,
```

```
% meme somme sur les deux diagonales :
```

```
X11 + X22 + X33 #= Sum,
```

```
X31 + X22 + X13 #= Sum,
```

```
% contrainte pour éviter réflexion horizontale
```

```
X11 #< X31,
```

```
% contrainte pour éviter réflexion verticale
```

```
X11 #< X13,
```

```
% contrainte pour éviter réflexion diagonale
```

```
X31 #< X13,
```

```
% contrainte pour éviter réflexion antidiagonale
```

```
X11 #< X33,
```



carre(X).

X =

2	7	6
9	5	1
4	3	8

?- carre(X).

Examples▲ History▲ Solutions▲ ☐ table results **Run!**

JOURNÉES MATH-SCIENCES

Trisième étude de cas: Sudoku

```
:- use_rendering(table).
:- use_module(library(clpfd)).

sudoku(Rows) :-
    % La valeur pour chaque "case" est entre 1 et 9
    append(Rows, Vs), Vs ins 1..9,
    % Toutes les valeurs dans une ligne sont différentes
    maplist(all_distinct, Rows),
    % Toutes les valeurs dans une colonne sont différentes
    transpose(Rows, Columns),
    maplist(all_distinct, Columns),
    % Toutes les valeurs dans les 9 blocs de 3x3 sont différentes
    Rows = [A,B,C,D,E,F,G,H,I],
    blocks(A, B, C),
    blocks(D, E, F),
    blocks(G, H, I).

blocks([], [], []).
blocks([A,B,C|Bs1], [D,E,F|Bs2], [G,H,I|Bs3]) :-
    all_distinct([A,B,C,D,E,F,G,H,I]),
    blocks(Bs1, Bs2, Bs3).
```

Problem =

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



Problem = [[_,_,_,_,_,_,_,_,_], [_,_,_,_,3,_,8,5], [_,_,1,_,2,_,_,_,_],
[_,_,5,_,7,_,_,_,_], [_,_,4,_,_,_,1,_,_], [_,9,_,_,_,_,_,_,_], [5,_,_,_,_,_,_,_,_],
,7,3], [,_,2,_,1,_,_,_,_], [_,_,_,_,4,_,_,9]], sudoku(Problem).

Problem =

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