

# UGMT41 : aide au projet

Pierre Cagne

# Description du jeu de sudoku

- **Données** : une grille 9 par 9 (avec certaines cases déjà remplies)
- **Remplissage** : chaque case peut contenir un chiffre 1, 2, ... ou 9.
- **Règles** :
  - toute case doit contenir un chiffre
  - chaque colonne contient un 1, un 2, ... et un 9
  - chaque ligne contient un 1, un 2, ... et un 9
  - chaque bloc contient un 1, un 2, ... et un 9
- **Solution** : la sélection d'un remplissage de chaque case, de sorte que chacune des règles est satisfaite une unique fois.

# Première traduction

- Problème : une *grille* avec remplissages possibles de cases, règles
- Solution : une *sélection* de remplissages pour chaque case tels que toute règle est satisfaite une unique fois



- Problème : un *triplet*  
 $(P, C, S)$
- Solution : une *sélection*  
 $p_0, p_1, \dots, p_n$   
d'éléments de  $P$  tels que toute contrainte est satisfaite par un unique  $p_i$

## Deuxième traduction

- Problème : un *triplet*

$$(P, C, S)$$

- Solution : une *sélection*

$$p_0, p_1, \dots, p_n$$

d'éléments de  $P$  tels que

toute contrainte est satisfaite  
par un unique  $p_i$



- Problème : une *matrice*

$$M = (m_{i,j}) \quad m_{i,j} \in \{0, 1\}$$

- Solution : une *sélection*

$$l_0, l_1, \dots, l_n$$

de lignes de  $M$  tels que

toute colonne contient un  
unique 1 parmi les lignes  $l_i$

# Algorithme récursif

Une solution

$$p_0, p_1, \dots, p_n$$

de  $(P, C, S)$  est exactement la donnée

- de  $c \in C$  et  $p_i$  avec  $p_i S c$
- et d'une solution

$$p_0, p_1, \dots, \hat{p}_i, \dots, p_n$$

de  $(P', C', S)$  où

$$P' = \{p : \forall c, p_i S c \Rightarrow p \notin c\}$$

$$C' = \{c : p \notin c\}$$

Une solution

$$l_0, l_1, \dots, l_n$$

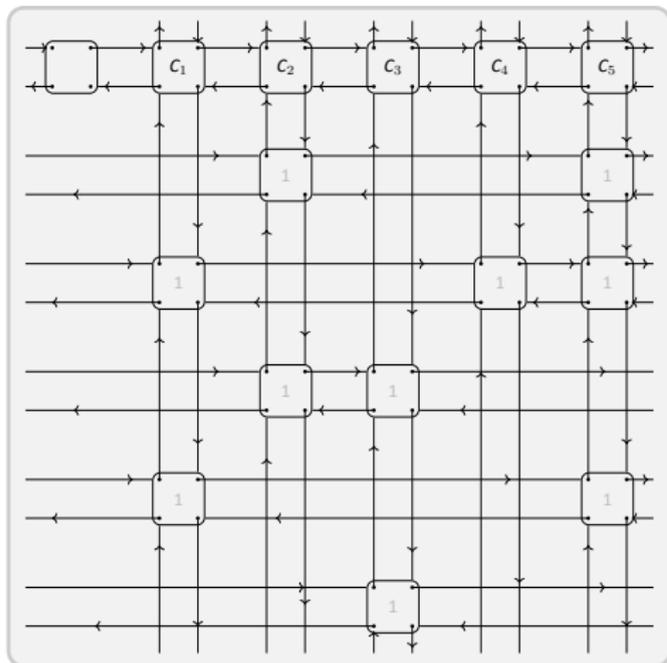
de  $M = (m_{i,j})$  est exactement la donnée

- de  $k$  et  $l_i$  avec  $m_{l_i, k} = 1$
- et d'une solution

$$l_0, l_1, \dots, \hat{l}_i, \dots, l_n$$

de la sous-matrice  $M'$  de  $M$  après suppression des lignes ayant un 1 au même colonne que la ligne  $l_i$   
suppression des colonnes ayant un 1 dans la ligne  $l_i$

# Un mot sur l'implémentation



```
typedef struct node_s node_t;
struct node_s {
    node_t *left, *right,
           *up, *down;
    /* etc. */
};
```

```
typedef struct column_s column_t;
struct column_s {
    node_t head;
    column_t *left, *right;
    /* etc. */
};
```