

## Décomposition algorithmique

### Programme principal

```
procédure pentree_valeur(sortF valeur, sortF numLigne, sortF numCol) c'est // trois
paramètres de sortie
procédure paffiche_grille(sortF grille) c'est // affiche la grille remplie au fur et à mesure
procédure pinit_grille(entF/sortF grille) c'est //initialise la grille avec des -1
fonction fverif(entF grille) délivre booléen c'est // vérifie les lignes, colonnes et blocs de la
grille
fonction fgrille_completee(entF grille) délivre booléen c'est // vérifie si il reste pas de -1

programme sudoku c'est
constante entier VALEUR_MIN := 0;
constante entier VALEUR_MAX := 8;
début
  grille : tableau[9][9] de entier;
  numLigne : entier;
  numCol : entier;
  estFini : booléen;
  valeur: entier;
  boucleL : entier;
  boucleC : entier;
  estFini := false;
  boucleL := 0;
  boucleC := 0;
  pinit_grille(entE/sortE grille,entE boucleL, entE boucleC);//
  tant que ((estFini != true )) faire
    paffiche_grille(sortE grille);
    pentree_valeur(sortE valeur, sortE numLigne, sortE numCol);
    si ((fverif(entE grille)))alors
      || grille[numLigne][numCol] := valeur;
      finsi
    si ((fgrille_completee(entE grille)) alors
      || estFini := true;
      finsi
    finfaire
  fin
```

## Liste procédures et fonctions

### procédure *paffiche\_grille*

affiche la grille la première fois (avec l'exemple de la maquette) et puis la grille remplie au fur à mesure avec les valeurs inscrites

### paramètre

*grille*(sortie):la grille dans son ensemble

---

### procédure *pinit\_grille*

initialise la grille avec des valeurs -1 la ou les cases sont vide(sans numéros), c'est à dire que le tableau sera rempli de -1 pour montrer les cases vides et ainsi faciliter la vérification de la fonction *fgrille\_completee*.

### paramètres

*grille*(entrée/sortie) : la grille dans son ensemble

*boucleL*(entrée) : variable de boucle pour les lignes

*boucleC*(entrée):variable de boucle pour les colonnes

---

### procédure *pentree\_valeur*

affiche une interface utilisateur permettant de saisir le numéro de la ligne, la valeur et le numéro de la colonne. La procédure vérifie la conformité des valeurs.

### paramètres

*valeur*(sortie) :la valeur saisie par l'utilisateur ou l'utilisatrice

*numLigne*(sortie):le numéro de la ligne saisi

*numCol*(sortie) : le numéro de la colonne saisi

---

### fonction *fverif*

vérifie les lignes, les colonnes, les blocs de la grille si les numéros sont à leur place.

### paramètre

*grille*(entrée):la grille dans son ensemble

### résultat

délivre un booléen, *false* si des valeurs ne sont pas correctes et *true* dans le cas contraire et

---

### fonction *fgrille\_completee*

la grille étant remplie de -1 la fonction vérifie si il reste des -1 dans la grille

### paramètre

*grille*(entrée) : la grille dans son ensemble

### résultat

un booléen pour savoir si la grille est remplie et donc complétée