# TP Programmation par contraintes

## Exercice 01 : N-Reines

Le problème des N Reines consiste à placer N reines d'échecs sur un échiquier N×N de telle sorte qu'aucune reine ne puisse attaquer une autre reine. Aux échecs, une reine peut attaquer toutes les pièces qui se trouvent soit sur la même ligne, soit sur la même colonne, soit sur la même diagonale.

1. Représenter le problème des quatre reines sous la forme d’un CSP
2. Écrire une fonction qui retourne le CSP de problème de N-reine
3. A l’aide de la fonction précédente, écrire une fonction qui affiche le CSP de problème de N-reine. L’Affichage doit être comme suit :
4. Écrire une fonction permettant de vérifier la consistance d'une assignation après l’ajout d’une nouvelle assignation
5. A l’aide de la fonction précédente testez la consistance des assignations suivantes
	1. ({},{'V2':1})
	2. ({'V1':2,'V2':4},{'V3':3})
	3. ({'V1':2,'V3':3},{'V2':4})
	4. ({'V1':1,'V2':4,'V4':3},{'V3':1})
6. Implémenter l'algorithme de **back-tracking** suivant :

**function** BACKTRACK(assignment, csp) **return** a solution or failure

    **if** assignment is complete **then** **return** assignment

    var ← SELECT-VARIABLE(var, assignment, csp)

    **for** **each** value **in** DOMAIN-VALUES(var, assignment, csp) **do**

        **if** value is consistent with assignment **then**

            add {var=value} to assignment

            result ← BACTRACK (assignment, csp)

            **if** result ≠ failure **then** return result

            remove {var=value} from assignment

    **return** failure

1. Tester le fonctionnement de l’algorithme pour n=3, n=4, n=8 et n=10
2. Ecrire fonction permettant d’afficher une solution du problème sur un graphe

## Exercice 02 : Coloriage de la carte de l'Australie

Il s'agit de colorier la carte de l’Australie, de sorte que deux régions ayant une frontière en commun soient coloriées avec des couleurs différentes. On dispose pour cela des 3 couleurs suivantes : bleu, rouge et vert.



Figure  : La carte de l'Australie

1. Modéliser ce problème sous la forme d'un CSP
2. Donner une structure donnée permettant de modéliser une carte géographique, puis donner la structure correspondante à la carté donné dans l'exercice
3. Écrire une fonction permettant de vérifier la consistance d'une assignation après l’ajout d’une nouvelle assignation
4. A l’aide de la fonction précédente testez la consistance des assignations suivantes
	1. ({},{"Q":'red'})
	2. ({'Q':'red'},{"NT":'red'})
	3. ({'Q':'red'},{"NT":'green'})
5. Implémenter l'algorithme de **back-tracking**:
6. Tester le fonctionnement de l’algorithme