

2020-2021

Algorithme N rênes



MEJRI Hazem

INSTITUT SUPERIEUR D'INFORMATIQUE ET DES TECHNIQUES DE COMMUNICATION – HAMMAM SOUSSE

Ce problème est de trouver un arrangement de reines N sur un échiquier, de sorte qu'aucune reine ne peut attaquer d'autres reines sur le conseil.

Les reines d'échecs peuvent attaquer dans n'importe quelle direction de manière horizontale, verticale, horizontale et diagonale.

Une matrice binaire est utilisée pour afficher les positions de N Queens, où aucune reine ne peut attaquer d'autres reine

Dans mon travaille j'ai utilisé deux fonction principaux :

- ✓ La première fonction intitulée chercher qui prend comme paramètre i,j qui sont les indices ligne , colonne cette fonction cherche si'il existe une reine ou la la casse et interdite en diagonales , lignes et colonnes la fonction retourne valeur boolean false si la case est libre , true si la case est interdite ou occupe
- ✓ La deuxième fonction intitulée solution qui prend comme paramètre la taille du matrice tout d'abord on teste si la taille de matrice est null la fonction prend true sinon on parcourt chaque case et on appelle la fonction chercher pour tester les diagonales et la ligne et la colonne , et on teste la la casse si elle est occupé déjà par une reine ou pas , si tous est bon la case prend '1' comme valeur et on appelle la même fonction Solution et on affecte comme paramètre n-1 et la teste si elle est true sinon la case prend '0' et la fonction prend false

Algorithme :

Fonction chercher(int i , int j)

Debut

Pour k de 0 à n faire

Si chess_board[i][k]== 1 || chess_board[k][j]==1

return True

Pour l de 0 à n faire

Pour l de 0 à n faire

Si (k+l==i+j) or (k-l==i-j)

Si chess_board[k][l]==1:

return True

return False

Fin

Fonction Solution(n)

Debut

Si n==0

Return True

Pour i de 0 à n faire

Pour j de 0 à n faire

Si (not(chercher(i,j))) and (chess_board[i][j]!=1)

Debut

chess_board[i][j] = 1

Si Solution(n-1)==True

return True

chess_board[i][j] = 0

Fin Si

Fin pour

Fin pour

Return false

Fin

Ecrire('Donner la taille du matrice')

Lire(n)

Int chess_board[n][n]

Solution(n)

Pour i de 0 à n faire

Pour j de 0 à n faire

Ecrire(chess_board[i][j])

Fin pour

Fin pour