Voici un troisième code qui utilise une méthode de résolution récursive. Il place une reine sur chaque ligne, en vérifiant si elle attaque une autre reine déjà placée sur la même colonne ou diagonale. Si une position est trouvée pour chaque ligne, la solution est retournée, sinon la fonction est appelée récursivement pour trouver une nouvelle position pour la reine sur la ligne actuelle. Cette approche utilise une technique appelée "depth-first search" (DFS) pour explorer toutes les solutions possibles jusqu'à ce qu'une solution valide soit trouvée.

Voici un quatrième code qui utilise l'algorithme génétique pour résoudre le problème des N-Reines. L'algorithme génétique est une technique d'optimisation qui s'inspire de l'évolution biologique pour trouver des solutions à des problèmes complexes. Il consiste à générer une population de solutions aléatoires, puis à les faire évoluer à travers des opérations de sélection, de croisement et de mutation, pour obtenir des solutions de plus en plus adaptées au problème.

Dans ce code, nous représentons chaque solution comme une liste de N entiers, où chaque entier représente la position de la reine sur la ligne correspondante. Nous initialisons une population de solutions aléatoires, puis nous évaluons chaque solution en comptant le nombre de conflits. Nous sélectionnons ensuite les meilleures solutions pour le croisement, en utilisant une roulette de sélection pondérée par le score de chaque solution. Nous croisons ensuite deux solutions parents pour créer deux nouvelles solutions enfants, en utilisant une méthode de croisement en deux points. Nous mutons ensuite les solutions enfants en inversant aléatoirement deux positions. Nous répétons ce processus pendant un certain nombre d'itérations, en conservant les meilleures solutions de chaque génération.