

TD3 Programmation Par Contraintes 3A-STI INSA Centre Val de Loire

Implémentation d'algorithmes de résolution

Nous allons réaliser plusieurs algorithmes (dans un premier temps Generate and Test puis backtrack) et comparer leur performance en terme de vitesse, pour la résolution du problème des n reines.

Pour mesure cette performance, on lancera au début du programme un chrono :

```
long start = System.currentTimeMillis();
```

et à la fin du programme :

```
System.out.println("Temps d'exécution : "+(System.currentTimeMillis()-start));
```

1) On souhaite réaliser l'algorithme *Generate and Test* pour résoudre un problème générique de n reines. Les contraintes à tester sont :

a) quelque soit i, j $X_i < > X_j$

b) quelque soit i, j $X_{i+i} < > X_{j+j}$

c) quelque soit i, j $X_{i-i} < > X_{j-j}$

Dans un premier temps, faites une fonction qui vérifie si une affectation totale est une solution ou non (retourne 1 si c'est le cas et 0 sinon)

Implémentez ensuite l'algorithme donné dans le cours, en utilisant cette fonction.

Jusqu'à quelle valeur de n pouvez vous monter pour un résultat donné en moins d'une seconde ? Tracez la courbe temps d'exécution fonction de n .

2) Idem pour l'algorithme *Simple Backtrack*. Il faudra cependant implémenter également une fonction qui prend une affectation partielle et vérifie si elle viole les contraintes ou non.