

Pour ce tp, j'ai fait la solution imp et la solution ppc, mais j'ai aussi fait une fonction de résolution itérative. Voici comment les trouver dans le code :

- La solution imp réursive (fonction : solve() qui appelle solve(int i) dans GraphColoringIMP.java)
- La solution itérative que j'ai faite en plus (fonction solveIte() dans GraphColoringIMP.java)
- La solution ppc avec chocoSolver (fonction solve(String nomFichier) dans GraphColoringPPC.java)

Il y a une fonction de vérification dans la classe GraphColoringIMP qui fonctionne avec Dataset. Cependant, j'ai fait une autre fonction de vérification dans GraphColoringPPC.java, (public boolean checkSolution (String nomFichier, int [] tabBis)), qui vérifie si le tableau passé en paramètre respecte bien les contraintes. Pour ça, je crée donc un nouveau model avec les valeurs du tabBis et je regarde si le ppc trouve une solution.

Dataset utilise BufferedReader, et crée deux tableaux, correspondants chacun à la colonne droite, puis gauche, des fichiers .col. Evidement, j'évite les commentaires, en allant directement à la ligne commençant par "@".

Pour afficher le résultat avec la correspondance des régions et des couleurs, il y a la fonction toStringRegions().