
Travail pratique

Heuristiques pour la coloration de graphes

Objectifs

Dans ce travail pratique vous devez programmer, tester et comparer deux heuristiques de coloration séquentielle des sommets d'un graphe. Les deux algorithmes que vous devez mettre en œuvre sont l'heuristique SL de Matula, Marbel et Isaacson que vous avez étudiée dans le travail théorique précédent ainsi qu'une simple coloration gloutonne dans l'ordre lexicographique.

Travail à effectuer

Vous devez réaliser un ou plusieurs programmes capables de lire un graphe simple à partir d'un fichier de données, d'effectuer une coloration gloutonne dans l'ordre croissant de numérotation des sommets ou d'appliquer l'heuristique SL et d'écrire la coloration calculée dans un fichier solution. Les formats des fichiers d'entrée et de sortie sont précisés plus loin dans la section correspondante.

Pour stocker chaque graphe, vous utiliserez un tableau de listes d'adjacence ou un tableau compact de sommets adjacents mais en aucun cas une matrice d'adjacence.

Pour la coloration, vous vous baserez sur votre étude théorique de l'heuristique SL et ne dévierez de votre travail que pour corriger d'éventuelles erreurs qu'il pourrait contenir ou pour remplacer une structure de données inadéquate par une autre plus adaptée..

Vous rédigerez un document qui contiendra en particulier la présentation et la discussion des résultats obtenus. Vous comparerez les deux méthodes sur chacun des jeux de données de l'archive

`//eistore1/profs/JHH/SIO-1718/TP1/DATA.zip.`

Outre le nombre de couleurs utilisées par chaque heuristique et pour chaque graphe vous présenterez également une analyse des temps d'exécution de chaque méthode.

Une marche à suivre pour utiliser vos exécutables sera fournie soit dans le document précédent sans donc un document spécifique.

Format des jeux de données et des solutions

Chaque graphe est stocké dans un fichier texte respectant le format qui suit.

- ▷ La première ligne contient deux entiers : le nombre n de sommets suivi du nombre m d'arêtes du graphe. Les sommets sont numérotés de 1 à n .
- ▷ Les m lignes suivantes définissent les arêtes du graphe et contiennent chacune deux entiers correspondant aux numéros des extrémités de l'arête.

TABLE 1 – Exemple de fichier de données et de fichier de solution.

Fichier de données (cycle sur 5 sommets)	Fichier solution (3-coloration optimale)
5 5	5 3
1 2	1 1
1 5	2 2
2 3	3 1
3 4	4 2
4 5	5 3

- ▷ Les arêtes apparaissent dans l'ordre lexicographique croissant. Chaque arête n'apparaît qu'une seule fois et ses extrémités sont forcément différentes.

Vos programmes devront produire des fichiers textes contenant les colorations obtenues et respectant le format qui suit.

- ▷ La première ligne contient deux entiers : le nombre n de sommets suivi du nombre k de couleurs utilisées.
- ▷ Les n lignes suivantes contiennent chacune deux entiers correspondant au numéro d'un sommet suivi de la couleur qui lui a été attribuée.
- ▷ Les sommets devront apparaître dans l'ordre croissant de leur numéro et les couleurs seront modélisées par des entiers consécutifs à partir de 1.

Les deux formats sont illustrés en table 1.

Modalités, documents à rendre et délais

- ▷ Le travail de programmation est à effectuer en C, en C++ ou en Java.
- ▷ Vos documents sont à envoyer à l'adresse jean-francois.heche@heig-vd.ch dans une archive **zip** dont le nom respectera impérativement le format

SIO_TP1_Nom_Prenom.zip

- ▷ Cette archive contiendra votre rapport au format PDF (dont le nom respectera le même format que l'archive mais avec une extension **.pdf**), le code suffisamment documenté de vos programmes, les exécutables correspondants et les fichiers des solutions obtenues pour chacun des jeux de données et chacune des heuristiques.
- ▷ Une version préliminaire de votre travail (rapport et code) devra être fournie **le lundi 9 avril 2018 avant minuit**.
- ▷ La version finale de votre travail devra être envoyée **au plus tard le jeudi 19 avril 2018 avant minuit**.