

Connexité dans les graphes

Florian Bourse

Dans ce TP, on va étudier des graphes non-orientés issus de 2 réseaux sociaux :

`twitch_gamers.csv` contient un graphe de 168 114 sommets et 6 797 557 arêtes, où chaque sommet représente un utilisateur Twitch, et deux sommets sont reliés si les utilisateurs se suivent mutuellement.

`git_stargazers.csv` contient un graphe de 1 448 038 sommets et 2 985 781 arêtes, où chaque sommet représente un développeur qui a aimé un projet d'apprentissage automatique ou de développement web, et deux sommets sont reliés si les utilisateurs se suivent mutuellement.

Les graphes sont donnés par une liste d'arêtes, les sommets étant numérotés de 0 à $n-1$, où n est le nombre de sommet. Le fichier `.csv` contient une ligne pour chaque arête, contenant les numéros des deux sommets joints, séparés par une virgule ",".

Les données ont été récupérées via <https://github.com/benedekrozemberczki/datasets>.

1 Composantes connexes d'un graphe

Question 1. Créer un fichier `exemple.csv` contenant un graphe à 10 sommets ayant 3 composantes connexes.

Nous allons implémenter une structure `unir` & trouver par deux tableaux de n entiers ou alors un tableau de $2n$ entiers.

Question 2. Créer une fonction `trouver` prenant en entrée un ou deux tableaux d'entiers et un entier i et qui renvoie le représentant de l'ensemble contenant i . Pour l'instant, on se passe du path-compression.

Question 3. Créer une fonction `unir` prenant en entrée un ou deux tableaux d'entiers et deux entiers i et j et qui met à jour le ou les tableaux pour que i et j soient dans le même ensemble. Pour l'instant, on se passe de l'union par rang.

Question 4. Remplir le code de la fonction `main` pour trouver le nombre de composantes connexes. La procédure à suivre est la suivante : pour chaque arête, unir les ensembles des deux extrémités. À la fin, compter le nombre d'ensembles différents.

Question 5. Implémenter les optimisations d'union par rang et de path compression.

Question 6. Trouver le nombre de composantes connexes des 2 graphes donnés.

2 Composantes 2-connexes d'un graphe

On appelle *point d'articulation* un sommet qui augmente le nombre de composantes connexes si il est retiré, et *pont* une arête qui augmente le nombre de composantes connexes si elle est retirée. Les composantes 2-connexes d'un graphe sont les sous-graphes maximaux qui ne contiennent pas de point d'articulation.

Question 7. Montrer que si deux sommets sont dans la même composante 2-connexe si et seulement si il existe un cycle passant par ces deux sommets.

Question 8. Utiliser cette propriété pour créer un algorithme qui compte le nombre de composantes 2-connexes, et l'appliquer aux 2 graphes donnés.

Hint : On commencera par trouver une forêt couvrante du graphe.