

Tri topologique [gp03] - Exercice

Karine Zampieri, Stéphane Rivière

Unisciel  algoprogram  Version 21 mai 2018

Table des matières

1 Tri topologique	2
1.1 Tri topologique	2
1.2 Algorithmique	2
1.3 Tris topologiques linéaire	3

alg - Tri topologique (TD)



Mots-Clés Structures relationnelles ■

Requis Axiomatique objet, Modèles, Gestion des exceptions ■

Difficulté ●●○ (1 h) ■



Objectif

Cet exercice réalise le tri topologique d'un graphe orienté acyclique.

1 Tri topologique

1.1 Tri topologique



Définition

Un **tri topologique** d'un graphe orienté acyclique $G = (S, A)$ est un ordre linéaire des sommets de G tel que si G contient l'arc (u, v) , u apparaît avant v .



Remarque

Le tri topologique d'un graphe peut être vu comme un alignement de ses sommets le long d'une ligne horizontale tel que tous les arcs soient orientés de gauche à droite.

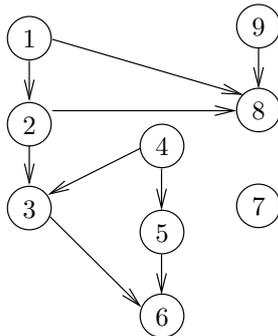


Attention

Le tri topologique d'un graphe orienté acyclique n'est pas forcément unique.



Proposez un tri topologique du graphe orienté acyclique de la figure suivante :



1.2 Algorithmique



Modifiez l'algorithme du parcours en profondeur (vu en cours) pour qu'il calcule pour chaque noeud u sa date de fin de traitement $ft[u]$ (la date à laquelle lui et ses fils ont été traités).



Quel lien pouvez-vous faire entre les dates de fin de traitement et un tri topologique ?



Dans le graphe de la première question, annotez les noeuds avec des fins de date de traitement lors d'un parcours en profondeur (« racines » successives) : noeuds 1, 9, 4 et 7.



Proposez un algorithme de tri topologique.



Quelle est sa complexité ?

1.3 Tris topologiques linéaire

Ce problème définit des tris topologiques de complexité linéaire.



Comment pouvez-vous améliorer votre algorithme pour qu'il soit de complexité linéaire ?



Écrivez l'algorithme correspondant.



Montrez que votre algorithme est de complexité linéaire.



Proposez un autre algorithme de tri topologique, basé cette fois-ci sur le fait qu'un sommet de degré entrant nul peut être placé en tête d'un tri topologique.



Justifiez votre algorithme.



Donnez sa complexité.