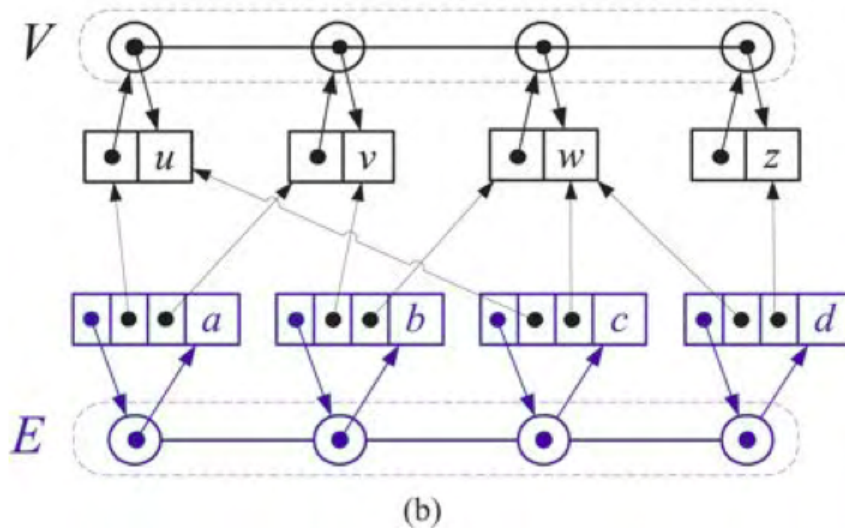
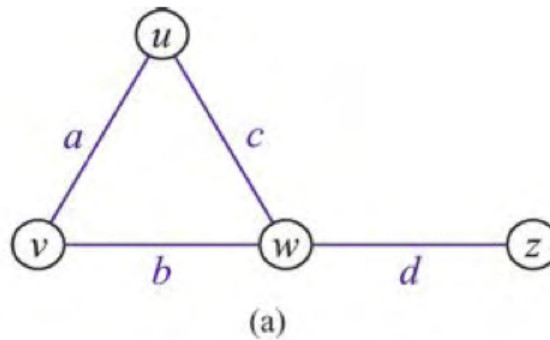


Algorithmique et structures de données : Mission 6

Groupe 1.2: Ivan Ahad - Jérôme Bertaux - Rodolphe Cambier
Baptiste Degryse - Wojciech Grynzel - Charles Jaquet

1 décembre 2014

Question 1 (Baptiste Degryse) Le graphe étant stocké dans une structure Edge List, il faut retrouver le Vertex dans une liste d'edges, en vérifiant à chaque fois si l'edge ne contient pas un pointer vers le vertex, si oui, il faut retirer l'edge. C'est une opération de complexité temporelle $O(m)$ parce qu'il faut toujours tout vérifier pour ne pas rater d'edge.



Edge List

source: Data Structures and Algorithms in Java Fourth Edition

RemoveEdge est de complexité $O(1)$ puisqu'il s'agit d'une liste doublement chaînée, tout comme insertVertex.

La réponse dépend bien de la structure de données utilisées pour mémoriser les contenus. Si les edges étaient stockées par vertex, il serait possible d'avoir de bien meilleures performances lors de l'opération removeVertex.

Le concept de location aware entry est indispensable pour avoir une complexité $O(1)$ pour la méthode `removeEdge`.

Question 2

Question 3

Question 4

Question 5

Question 6