Rapport de TP d'Algorithmique

Loris Croce

18 mars 2018

Table des matières

Chapitre 1

Gestion de partitions

1.1 TDA Gestion de partition

Le TDA *Gestion de partition* est un TDA qui qui définit une partition p composée d'éléments e [1], lesquels sont regroupés en son sein par classes. Il dispose de deux opérations :

— `p.classe(e)` : retourne la classe de l'élément e.

— `p.fusion(c1, c2)` : fusionne les classes c_1 et c_2 de la partition p.

Pour implémenter ce TDA une *interface* `GestionPartition` qui définit la structure du TDA a été créée. Deux solutions ont été implémentées : Le *Tableau de partition* et le *Tableau de pères*.

1.1.1 Tableau de partition

Comme les éléments gérés ici sont des entiers, il est possible de les représenter sous la forme d'indices d'un tableau où les cases contiendront la classe dans laquelle chaque élément est stocké. La classe associée à cette implémentation est `TableauPartition`.

Les complexités associées à cette implémentation sont :

— `p.classe(e)` : $\mathcal{O}(1)$, ici implémentée par `getClasse(int e)`.

— `p.fusion(c1, c2)` : $\mathcal{O}(n)$, ici implémentée par `fusion(int c1, int c2)`.

1.1.2 Tableau de pères

Cette implémentation est associée à celle par *forêts* en la traduisant sous forme de tableau où comme pour le tableau de père les indices représentent les éléments mais les valeurs des cases renseignent le *père* de l'élément [2]. La classe associée à cette implémentation est `TableauPerePartition`.

1. Ici, par souci de simplicité les éléments seront des entiers
2. si cet élément est racine on considèrera qu'il est son propre père.

Les complexités associées à cette implémentation sont :

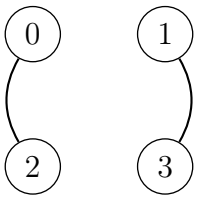
— `p.classe(e)` : $\mathcal{O}(h)$, ici implémentée par `getClasse(int e)`.

— `p.fusion(c1, c2)` : $\mathcal{O}(1)$, ici implémentée par `fusion(int c1, int c2)`.

1.2 Composantes connexes

Pour résoudre le problème des composantes connexes il a été nécessaire de créer une classe `Graphe` contenant les arêtes sous la forme d'une matrice d'entiers et les sommets sous la forme d'un tableau d'entiers. Il a également fallu surcharger le constructeur de `TableauPartition` pour qu'il prenne en paramètre un type `Graphe` et effectue les fusions nécessaires pour que les sommets reliés appartiennent à la même classe, montrant ainsi les composantes connexes du graphe.

Exemple Un graphe G de la forme :



1.3 Arbres couvrants minimaux

Chapitre 2

Programmation dynamique

2.1 Recherche de la distance d'édition