

INFO-F203 - Algorithmique 2: Projet 1

Rapport

Pascal Tribel
Noémie Muller

8 novembre 2018

Sommaire

I	Sous-arbre de poids maximum	1
1	Introduction	2
2	Choix d'implémentation	2
2.1	La classe "Tree"	2
2.1.1	La méthode somme	2
2.2	La fonction maxSubTree	2
3	Complexité	2
3.0.1	Somme	2
3.0.2	Algorithme maxSubTree	2
4	Conclusion	2
II	Les hypergraphes et hypertrees	3
5	Introduction	3
6	Choix d'implémentation	3
6.1	Les classes	3
6.1.1	La classe HyperNode	3
6.1.2	La classe HyperArete	3
6.1.3	La classe HyperGraphe	3
6.1.4	La classe Node	3
6.1.5	La classe Graphe	3
6.2	Les fonctions	3
6.2.1	incidenceGraph	3
6.2.2	primalGraph	3
6.2.3	dualGraph	3
6.2.4	isCordal	3
6.2.5	getMaximalClique	3
6.2.6	getCycles	3
6.2.7	isAlphaAcyclic	3
6.2.8	isHyperTree	3
7	Complexité	3
7.1	Séparément	3
7.1.1	incidenceGraph	3
7.1.2	primalGraph	3
7.1.3	dualGraph	3

7.1.4	isCordal	3
7.1.5	getMaximalClique	3
7.1.6	getCycles	3
7.1.7	isAlphaAcyclic	3
7.2	L'algorithme final	3
8	Conclusion	3

Première partie

Sous-arbre de poids maximum

1 Introduction

2 Choix d'implémentation

2.1 La classe "Tree"

2.1.1 La méthode somme

2.2 La fonction maxSubTree

3 Complexité

3.0.1 Somme

3.0.2 Algorithme maxSubTree

4 Conclusion

Deuxième partie

Les hypergraphes et hypertrees

5 Introduction

6 Choix d'implémentation

6.1 Les classes

6.1.1 La classe HyperNode

6.1.2 La classe HyperArete

6.1.3 La classe HyperGraphe

6.1.4 La classe Node

6.1.5 La classe Graphe

6.2 Les fonctions

6.2.1 incidenceGraph

6.2.2 primalGraph

6.2.3 dualGraph

6.2.4 isCordal

6.2.5 getMaximalClique

6.2.6 getCycles

6.2.7 isAlphaAcyclic

6.2.8 isHyperTree

7 Complexité

7.1 Séparément

7.1.1 incidenceGraph

7.1.2 primalGraph

7.1.3 dualGraph

7.1.4 isCordal

7.1.5 getMaximalClique

7.1.6 getCycles

7.1.7 isAlphaAcyclic

7.2 L'algorithme final

8 Conclusion