Sorbonne Université 2021 - Master d’informatique

COMPLEX - Complexité, algorithmes randomisés et approchés

**Couverture de graphe**

Almehdi KRISNI , Alessia LOI

Groupe 3

Dans ce projet, nous avons choisi d’adopter le langage Python pour implémenter les méthodes demandées.

**Notation utilisée dans ce rapport**

  : graphe non orienté

  : ensemble de sommets de

  : ensemble des arêtes de

: couplage de , i.e. ensemble d’arêtes n’ayant pas d’extrémité en commun

: couverture de , i.e. ensemble de sommets tel que toute arête

a au moins une de ses extrémités dans

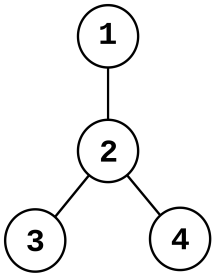
**Partie 2 : graphes**

Dans notre conception, un graphe est représenté par un dictionnaire = dont :

* chaque nœud du graphe est une clé
* pour chaque clé , la valeur est une liste de sommets adjacents au sommet de la clé.

Exemple. Soit un graphe , avec = =

Le dictionnaire représentant le graphe sera : =



Nous nous servons de la librairie Python « networkx » apte à la création et manipulation de graphes, mais nous ne l’utilisons que pour l’affichage graphique des graphes.

**Partie 3 : méthodes approchées**

|  |
| --- |
| 3.1 On rappelle que l’algorithme algo-couplage est 2-approché.  Montrer que l’algorithme glouton précédent n’est pas optimal.  En déduire que l’algorithme glouton n’est pas -approché, pour un certain à préciser. |

|  |
| --- |
| 3.2 Comparaison des méthodes algo-couplage et algo-glouton  en fonction de et du point de vue de leur temps de calcul et de la qualité des soltions. |

**Partie 4 : séparation et évaluation**