

Structures de données et algorithmes

Répétition 9: Graphes

Jean-Michel BEGON – Romain MORMONT
<http://www.montefiore.ulg.ac.be/~jmbegon>

18 mai 2018

Exercice 1

- (a) A quoi ressemble un graphe dont tous les sommets sont de degré 1 exactement ?
- (b) A quoi ressemble un graphe dont tous les sommets sont de degré 2 exactement ?
- (c) Si un graphe non orienté possède n sommets, tous de degré d , combien comporte-t-il d'arêtes ?

Exercice 2

Soit un graphe G dont A est la matrice d'adjacence.

- (a) Caractériser le graphe G' dont A^T est la matrice d'adjacence.
- (b) Quelle interprétation peut-on faire du produit AA ?
- (c) Quelle interprétation peut-on faire du produit A^k ?

Exercice 3

Pour les cas suivants, est-il plus approprié d'utiliser un graphe implémenté par une liste d'adjacences ou par une matrice d'adjacence ? Justifier.

- (a) Le graphe possède 10.000 sommets et 20.000 arêtes et on souhaite minimiser l'espace mémoire utilisé.
- (b) Le graphe possède 10.000 sommets et 2.000.000 arêtes et on souhaite minimiser l'espace mémoire utilisé.
- (c) On souhaite déterminer le plus rapidement possible si deux sommets sont adjacents, peu importe l'espace mémoire requis.

Exercice 4

Illustrer graphiquement l'exécution de l'algorithme de Dijkstra sur le graphe de la Figure 1. On souhaite trouver tous les plus courts chemins à partir du sommet BWI.

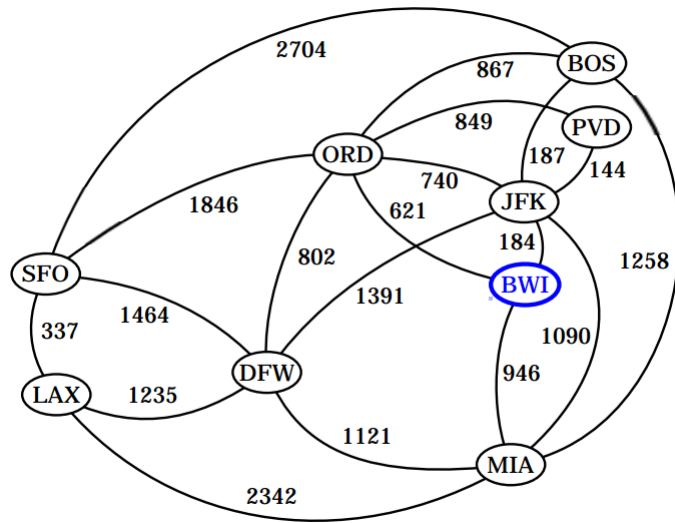


FIGURE 1 – Graphe

Exercice 5

Dans certaines applications (les réseaux notamment), une architecture de graphes est utilisée et un des nœuds joue souvent un rôle spécial par rapport aux autres (e.g., un serveur de fichiers au sein d'un réseau d'ordinateurs). Pour obtenir des performances optimales, on souhaiterait placer ce nœud au "centre" du graphe. Pour cela, on définit, étant donné un graphe G et un nœud v , l'excentricité de v comme la longueur du plus long des plus courts chemins entre v et tous les autres nœuds de G . Le centre de G est défini comme le nœud d'excentricité minimale.

- Proposer un algorithme pour calculer le centre d'un graphe G .
- Quelle est sa complexité si G est implémenté par une liste d'adjacences ?
- Quelle est sa complexité si G est implémenté par une matrice d'adjacence ?

Exercice 6

Les échelles de mots est un jeu (inventé par Lewis Carroll) où l'on doit passer d'un mot à un autre en utilisant des mots intermédiaires, et où, à chaque étape, une seule lettre est enlevée, ajoutée ou remplacée par une autre, les autres restant identiques et dans la même position. Toutes les formes grammaticales sont permises. On se restreint ici aux mots de même longueur.

- Passer (en anglais) de LESS à MORE.
- Proposer un algorithme pour résoudre ce problème de façon générique, étant donné une liste de mots de n lettres.

Bonus

Islands

Huit îles se trouvent sur un lac. Le gouvernement local souhaite construire sept ponts pour les relier, de sorte qu'on puisse rejoindre n'importe quelle île depuis n'importe quelle autre en

traversant un ou plusieurs ponts. Le coût de construction d'un pont est proportionnel à sa longueur. Sachant que la table ci-dessous énumère les distances entre chaque île, quels sont les ponts à construire pour minimiser le coût total de construction ?

	1	2	3	4	5	6	7	8
1	-	240	210	340	280	200	345	120
2	-	-	265	175	215	180	185	155
3	-	-	-	260	115	350	435	195
4	-	-	-	-	160	330	295	230
5	-	-	-	-	-	360	400	170
6	-	-	-	-	-	-	175	250
7	-	-	-	-	-	-	-	305
8	-	-	-	-	-	-	-	-

Le 2-coloriage

Pour votre anniversaire, vous décidez d'organiser une petite fête. Après avoir soigneusement dressé la liste de vos invités, vous dessinez un graphe de connaissance à l'aide des réseaux sociaux. Le graphe est tel que chaque nœud $p_i \in V$ représente un invité et chaque arête $(p_i, p_j) \in E$ signifie que p_i et p_j se connaissent (le graphe est non-orienté).

Comment, à l'aide de ce graphe, partitionner vos invités en deux groupes V_r et V_b ($V_r \cup V_b = V$) tel qu'aucune personne ne connaisse aucune autre personne de son sous-groupe ($V_r \cap V_b = \emptyset$). Est-ce toujours possible ? Quelle méthode de résolution de problème est employée ?

Might & Magic

Lors de la quête d'archimage, *Might and Magic X - Legacy* met le joueur face à l'énigme régulière suivante :

Une pièce est composée de 3x3 cases, une porte d'entrée (e) et une porte de sortie (s)

```

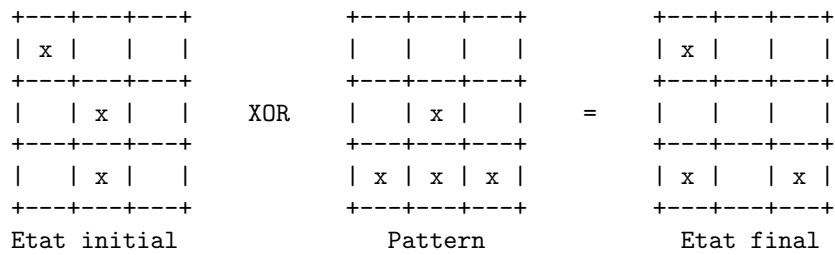
      s
+---+ / \ +---+
| 0 | 1 | 2 |
+---+---+---+
| 3 | 4 | 5 |
+---+---+---+
| 6 | 7 | 8 |
+---+ \ / +---+
      e

```

Initialement, la porte de sortie est fermée. Lorsqu'on marche sur une des cases, plusieurs d'entre elles s'allument et d'autres s'éteignent. Sortir de la pièce (par l'entrée) remet le puzzle dans son état initial. Finalement, les mouvements du joueur sont restreints à haut-bas-gauche-droite (pas de mouvements en diagonale).

Il semble logique de penser que la porte de sortie s'ouvrira lorsque toutes les cases seront allumées. C'est à dire, lorsque le joueur aura trouvé un chemin - passant potentiellement plusieurs fois sur certaines cases - tel qu'aucune case ne soit éteinte. Il ne semble pas nécessaire de terminer sur la case faisant face à la sortie, quoiqu'il s'agisse là d'une note de style incontestable.

Après quelques minutes de test, on se rend compte que chaque case a un pattern prédéfini, et que le résultat d'éclairage est un XOR entre ce pattern et l'état actuel de la pièce. Par exemple :



Un simulateur est présent à l'adresse http://www.montefiore.ulg.ac.be/~jmbegon/?sda2014_2015#enigmaTower.

- (a) Comment modéliser ce problème sous forme d'un graphe ?
 - (a) Que représente un nœud ? Combien y en a-t-il ?
 - (b) Que représente une arête ? Quels sont les degrés des nœuds ?
- (b) La solution à cette énigme référencée par un site de solution assez connu est la suivante :

↑, ↑, →, ←, ←, →, ↓, ↑, ↑, →, ←, ←, →, ↓, →, ↓, ←↑, ↓, ↑, ↑

Soit 21 mouvements. Cette solution est-elle optimale du point de vue du nombre de mouvements ? Quel algorithme faut-il utiliser pour résoudre ce problème ?

- (c) Dans quelle mesure le nombre de mouvements augmente-t-il si on ajoute la contrainte de terminer devant la porte de sortie ?
- (d) Est-ce que chaque état est accessible à partir de l'état initial ? Est-ce que l'état initial est accessible à partir d'au moins un autre état (sans repasser par la case reset).

Remarque : ce genre d'énigme est récurrent dans les jeux. Par exemple, dans *Bordelands 2*, une énigme similaire consiste à actionner 4 leviers aux patterns prédéfinis (à lieu de marcher sur les cases) afin d'obtenir 5 lumières vertes (au lieu d'illuminer les cases).