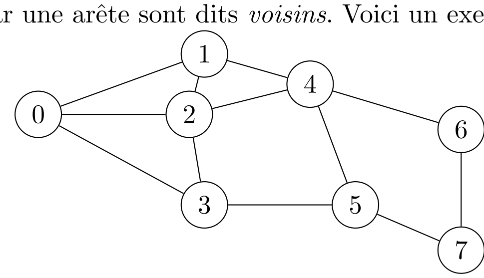
TP 2: Parcours de graphes

7 octobre 2013

Dans ce TP, nous allons travailler autour des représentations de graphes, puis implémenter les parcours en largeur et en profondeur.

1 Introduction

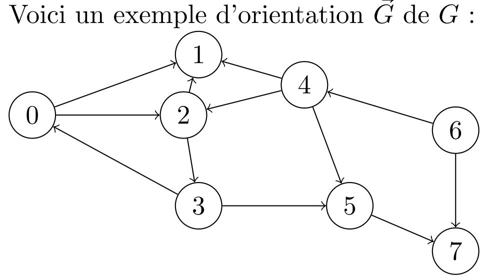
Un graphe non orienté est la donnée d'un ensemble de sommets S et d'un ensemble d'arêtes $A \subseteq \binom{S}{2}$ liant deux sommets entre eux. Deux sommets liés par une arête sont dits *voisins*. Voici un exemple de graphe G non orienté :



avec $S = \{0, 1, 2, 3, 4, 5, 6, 7\}$ et $A = \{\{0, 1\}, \{0, 2\}, \{0, 3\}, \{1, 2\}, \{2, 3\}, \{1, 4\}, \{2, 4\}, \{3, 5\}, \{4, 5\}, \{4, 6\}, \{5, 7\}, \{6, 7\}\}$.

Un graphe orienté est la donnée d'un ensemble de sommets S et d'un ensemble d'arcs $A \subseteq S \times S$ liant un sommet à un autre. Si (u, v) est un arc, u est un *prédecesseur* de v, et v est un *successeur* de u. L'orientation d'un graphe non orienté (S, A) est un graphe orienté (S, A') tel que $\{u, v\} \in A \Leftrightarrow [(u, v) \in A$ ou exclusif $(v, u) \in A]$.

Voici un exemple d'orientation $\vec{G}$ de G :



$A = \{(0, 1), (0, 2), (3, 0), (2, 1), (2, 3), (4, 1), (4, 2), (3, 5), (5, 7), (6, 4), (5, 7), (6, 7)\}$.

1.1 Représentation et génération

On rappelle que la matrice d'adjacence Adj d'un graphe orienté $G = (S, A)$ est une matrice carrée $|S| \times |S|$ telle que $Adj_{i,j} = 1$ si $(i, j) \in A$ et $Adj_{i,j} = 0$ sinon. On définit de la même manière la matrice d'adjacence d'un graphe non orienté. Notez que celle-ci est alors symétrique. Les listes d'adjacence d'un graphe non orienté (resp. orienté) $G = (S, A)$ correspondent à $|S|$ listes des voisins (resp. successeurs) de chaque sommet.

Exercice 1 *Écrire une fonction qui prend la matrice d'adjacence d'un graphe et dit s'il peut ou non s'agir d'un graphe non orienté.*

Exercice 2 *Écrire une fonction qui prend les listes d'adjacence d'un graphe et dit s'il peut ou non s'agir d'un graphe non orienté.*

Exercice 3 *Écrire une fonction qui prend la matrice d'adjacence d'un graphe et retourne le même graphe sous forme de listes d'adjacence.*

Exercice 4 *Écrire la transformation inverse : listes d'adjacence vers matrice d'adjacence.*

Exercice 5 *Écrire une fonction qui prend un graphe non orienté et qui renvoie une orientation aléatoire du graphe.*

Exercice 6 *Écrire une fonction qui prend un entier n et un réel $0 \leqslant p \leqslant 1$ et qui renvoie un graphe aléatoire à n sommets et ayant pour tout couple de sommets u, v, l'arc (u, v) avec probabilité p.*

1.2 Parcours

Ici, on peut supposer travailler avec des graphes orientés.

Exercice 7 *Implémenter le parcours en largeur. Votre programme prendra un graphe G sous la forme que vous voulez et un sommet v de ce graphe et devra retourner la liste ordonnée des sommets rencontrés par un parcours en largeur depuis v.*

Vous pouvez maintenant savoir si un graphe non orienté est connexe.

Exercice 8 *Implémenter le parcours en profondeur.*

Exercice 9 *Testez vos programmes sur des "gros" graphes.*