

**info505 : graphes et algorithmes**  
**TD 1 : représentations et parcours en largeur**

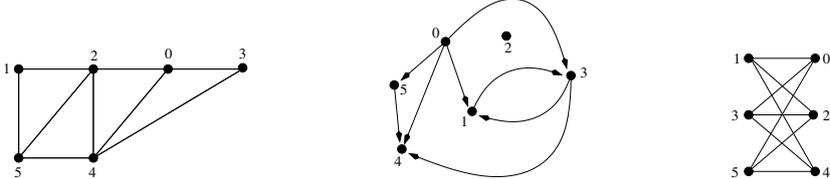
Pierre Hyvernât  
 Laboratoire de mathématiques de l'université de Savoie  
 bâtiment Chablais, bureau 22, poste : 94 22  
 email : Pierre.Hyvernât@univ-savoie.fr  
 www : <http://www.lama.univ-savoie.fr/~hyvernât/>

**Exercice 1 : représentations des graphes**

*Question 1.* Pour chacun des graphes suivants sur  $S = \{0, 1, 2, 3, 4, 5\}$ , donnez la représentation sous les différentes formes données en cours :

- liste d'adjacence,
- matrice d'adjacence.

Une arête non-orientée peut être vue comme deux arêtes orientées : une dans chaque direction...



*Question 2.* Donnez la représentation sous forme de matrice d'adjacence des graphes sur  $\{0, 1, 2, 3, 4\}$  représentés par les listes d'adjacence suivantes :

$\begin{bmatrix} - \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1, 2, 3, 4 \\ 0, 2, 4 \\ 0, 1, 3 \\ 0, 2, 4 \\ 0, 1, 3 \end{bmatrix}$	$\begin{bmatrix} 1, 2, 3, 4 \\ 0, 2, 3, 4 \\ 0, 1, 3, 4 \\ 0, 1, 2, 4 \\ 0, 1, 2, 3 \end{bmatrix}$
---	--	--

Dessinez les graphes correspondants et comparez vos résultats...

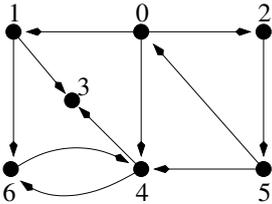
*Question 3.* Écrivez un algorithme qui permettra de passer de la représentation d'un graphe sous forme de liste d'adjacence à une représentation en matrice d'adjacence.

Faites un autre algorithme pour la transformation inverse.

Que se passe-t'il si on applique le premier algorithme puis le second ?

**Exercice 2 : parcours en largeur**

On va utiliser le graphe suivant



*Question 1.* Donner une représentation sous forme de liste d'adjacence de ce graphe.

*Question 2.* Utilisez l'algorithme décrit en cours pour faire un parcours en largeur du graphe. Décomposez l'algorithme pour être sûr de comprendre ce qui se passe.

*Question 3.* Dessinez la forêt obtenue par l'algorithme (dans le tableau `pere[]`).

*Question 4.* Est-ce que cette forêt est entièrement déterminée par le graphe? Par sa représentation?

Justifiez attentivement votre réponse.

*Question 5.* Ré-appliquez l'algorithme du cours en prenant, pour la première boucle (“pour tous les sommets  $u$ ”) les sommets dans l'ordre suivant : 3, 4, 5, 6, 0, 1, 2.

Dessinez la forêt correspondant à cette exécution. Qu'en pensez-vous?

*Question 6.* Calculez, en utilisant l'algorithme du cours, le tableau des distances au sommet 2.

### **Exercice 3 : deux applications**

*Question 1.* Le *diamètre* d'un graphe est le maximum des distances possibles entre deux sommets. Donnez un algorithme efficace permettant de calculer le diamètre d'un arbre non orienté et analysez sa complexité.

*Question 2.* Un graphe non orienté est *biparti* si on peut séparer ses sommets en deux sous-ensembles  $S_1$  et  $S_2$  qui ont la propriété suivante :

*il n'y a aucune arête entre deux sommets de  $S_1$  et il n'y a aucune arête entre deux sommets de  $S_2$ . (Autrement dit, toutes les arêtes sont incidentes à un sommet de  $S_1$  et un sommet de  $S_2$ .)*

Décrivez un algorithme efficace pour tester si un graphe (non-orienté connexe) est biparti. Donnez la complexité de cet algorithme.