

Examen Graphes et langages - Semestre 2 - 2017-2018

Le seul document autorisé pour l'examen est une feuille format A5 recto-verso manuscrite. Tous les autres documents sont interdits. Les calculatrices, ordinateurs, PDA, téléphones,... sont interdits. Les exercices peuvent être traités dans l'ordre de votre choix. La présentation et la rédaction sont des éléments d'appréciation des réponses. **Le sujet comporte une page.**

Exercice 1

On considère le graphe de la figure 1.

1. Donner un chemin de longueur 4 sans boucle.
2. Donner une boucle de longueur 6.
3. Donner la représentation mathématique du graphe(avec des ensembles V et E).
4. Donner sa représentation par matrice d'adjacence.
5. Dessiner sa représentation par liste d'adjacence.
6. Dessiner l'arbre obtenu par un parcours en largeur en partant de 1.

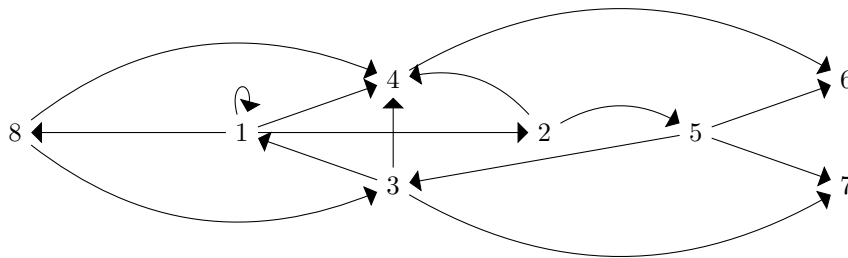


FIGURE 1 – Graphe Exercice 1

Exercice 2

On considère un ensemble fini de villes. On considère un ensemble E , formé de couples de V , défini par $(x, y) \in E$, s'il existe un bus qui part le matin de x en direction de y . On considère aussi l'ensemble F , formé de couples de V , défini par $(x, y) \in F$, s'il existe un bus qui part le soir de x en direction de y . On pose $G_1 = (V, E)$ et $G_2 = (V, F)$. Pour chacune des propriétés suivantes (indépendantes), donner une traduction dans le vocabulaire des graphes.

1. Il y a une ville d'où ne part aucun bus le matin.
2. Quand un bus part le matin, c'est toujours vers une autre ville.
3. Il n'y a aucune ville d'où ne part aucun bus le matin.
4. Il y a une ville vers laquelle aucun bus ne part le matin.
5. On peut toujours quitter n'importe quelle ville le soir.
6. Si un bus part le matin d'une ville vers une autre, il y a un bus le soir dans l'autre sens.

Exercice 3

Les algorithmes demandés dans cet exercice peuvent être présentés en Python avec la structure vue en TD ou en pseudo-code. Il est important de les décrire en français et d'utiliser un codage par listes d'adjacence.

1. Ecrire un algorithme qui compte le nombre de sommets qui ont au moins un voisin.
2. Ecrire un algorithme qui teste s'il existe au moins un sommet qui est voisin de lui-même.
3. Ecrire un algorithme qui étant donnés deux sommets p et q différents, teste s'il existe un chemin de longueur 2 de p à q .

Exercice 4

1. On considère l'arbre de la figure 2. Donner l'ordre des sommets obtenus lors d'un parcours préfixe (les fils sont explorés par ordre croissants de numéros).
2. Même question avec un parcours suffixe (idem pour l'ordre).
3. Donner un exemple d'application du parcours suffixe.

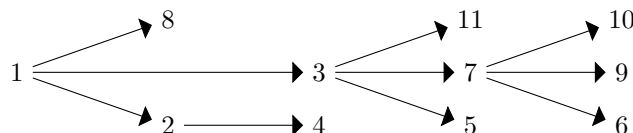


FIGURE 2 – Arbre Exercice 4