

Algorithmique avancée

Devoir surveillé

Les calculatrices ne sont pas autorisées.

Les exercices peuvent être traités dans le désordre.

La notation prendra en compte le soin et la clarté de la rédaction.

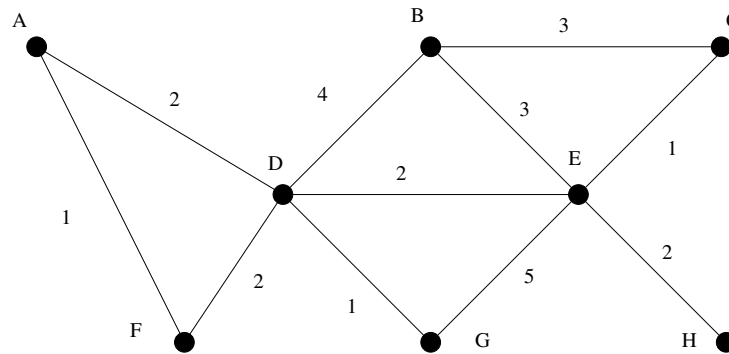


FIGURE 1 –

Exercice 1.

On considère le graphe de la figure 1.

1. Ecrire la matrice d'adjacence de ce graphe.
2. Appliquer l'algorithme de Prim pour trouver un arbre couvrant de poids minimal du graphe. La rédaction indiquera à chaque étape la ou les arêtes considérées et la décision prise.

Exercice 2.

L'algorithme de Dijkstra s'applique sur un graphe orienté exactement comme sur un graphe non-orienté, en ne considérant les arêtes que si elles sont prises dans le bon sens.

1. Appliquer cet algorithme au graphe de la figure 2 pour déterminer le coût minimal d'acheminement des marchandises de l'usine u à chacun des trois ports p_1, p_2, p_3 .
La rédaction indiquera dans quel ordre les sommets du graphe sont ajoutés à la solution partielle.

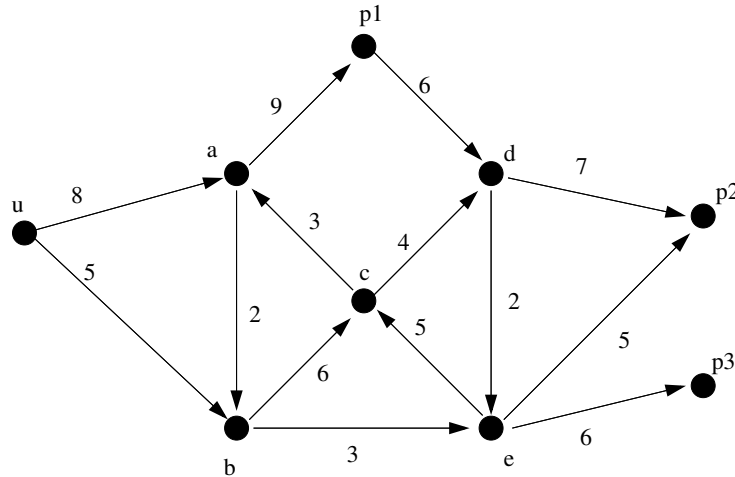


FIGURE 2 –

2. On suppose qu'un coût de passage c_v est affecté également à chaque sommet v , et que le coût total d'un chemin est alors égal à la somme des coûts des arêtes et des sommets traversés.
Proposer et justifier une modification de l'algorithme de Dijkstra pour trouver un chemin de coût minimal avec ce nouveau coût.

Exercice 3.

Un graphe non orienté est 2-colorable si ses sommets peuvent être coloriés en deux couleurs (disons rouge et bleu) de façon à ce qu'aucune arête ne relie deux sommets de la même couleur (cette notion est équivalente à celle de graphe biparti).

1. On considère un parcours DFS d'un graphe 2-colorable. Que dire de la couleur des sommets suivant leur niveau dans l'arbre?
2. Proposer un algorithme permettant de déterminer si un graphe est 2-colorable. Démontrer sa validité et donner sa complexité.

Exercice 4.

Le problème du voyageur de commerce ne peut être abordé que par des heuristiques, c'est-à-dire qu'on détermine des solutions approchées dont on ne

connaît pas la vraie distance à la solution exacte. Un des exercices de TD montrait qu'on pouvait par exemple établir une heuristique telle que la solution est au plus deux fois plus chère que la solution optimale.

Afin d'avoir une meilleure idée de la qualité de l'approximation obtenue par une heuristique, il peut également être intéressant d'avoir une borne inférieure sur la valeur de la solution optimale, ce qui est le but de cet exercice.

1. Énoncer le problème du voyageur de commerce sur un graphe G , dont on note w la fonction des poids des arêtes.
2. On considère le graphe H complet (toutes les arêtes sont présentes) ayant les mêmes sommets que G . Le poids d'une arête de H est le poids d'un plus court chemin dans G entre ses extrémités. On admet
 - (a) que la solution optimale sur H correspond à une solution optimale sur G .
 - (b) que la solution optimale C sur H est un cycle hamiltonien, c'est-à-dire que le voyageur ne repasse jamais deux fois par le même sommet.

A l'aide de quel algorithme construire le graphe H ?

3. Soit v un sommet quelconque du graphe H et H_v le graphe obtenu en supprimant v et toutes les arêtes adjacentes à v . On note
 - T_v un arbre de poids minimal de H_v
 - $c(v) = \min (w(e_1) + w(e_2) \mid e_1 \text{ et } e_2 \text{ adjacents à } v \text{ dans } H)$
 Soit C le trajet optimal. Montrer que, pour tout v ,

$$w(C) \geq w(T_v) + c(v)$$

4. En déduire un algorithme permettant de déterminer une borne inférieure au problème du voyageur de commerce sur G . Quelle est la complexité de cet algorithme ?