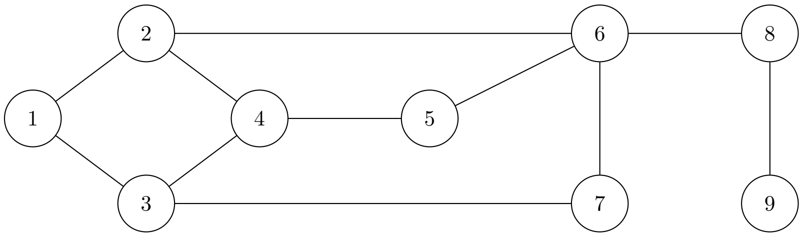
Module LU2IN003 Graphes : parcours en largeur semaine 10

Exercice(s)

Exercice 1 – Parcours en largeur d'un graphe non orienté

On considère le graphe non orienté $G_1 = (V_1, E_1)$:



Question 1
Pour chacun des parcours génériques de G_1 suivants, dire s'il est ou non un parcours en largeur : $(8,6,9,2,5,7,1,4,3)$, $(6,8,5,7,4,2,3,1,9)$, $(4,2,3,5,1,7,6,8,9)$, $(3,1,4,2,6,5,7,8,9)$

Justifier les réponses négatives.

Question 2
Donner trois parcours en largeur de G_1, l'un partant du sommet 1, un autre du sommet 9 et un troisième du sommet 5.

Question 3
On considère le parcours $L = (3,1,4,7,2,5,6,8,9)$ de G_1. Dire quel est le premier sommet ouvert de chaque sous-parcours de L. Le parcours L est-il un parcours en largeur ?

Exercice 2 – Parcours en largeur et distance

Soit $G = (V'E)$ un graphe non orienté connexe ayant n sommets. Soit $s \in V$ et $L = (s_1, \ldots, s_n)$ une liste des n sommets de G telle que $s_1 = s$ et $dist_s(s_1) \leq dist_s(s_2) \leq \ldots \leq dist_s(s_n)$.

Question 1
Prouver que L est un parcours générique de G.

Question 2
L est-elle nécessairement un parcours en largeur de G ? Justifier la réponse.

Exercice 3 – Graphe de liaison en largeur

Question 1
On considère le parcours en largeur $L = (4,2,3,5,1,6,7,8,9)$ du graphe G_1. Dessiner le graphe de liaison en largeur de L. Existe-t-il un autre graphe de liaison en largeur pour L ?

Question 2
Soit $G = (V, E)$ un graphe non orienté connexe et L un parcours en largeur de G. Montrer par contradiction que le graphe de liaison en largeur de L est unique.

Exercice 4

On rappelle l'algorithme de construction d'un parcours en largeur :

Require: Un graphe non orienté connexe $G = (V, E)$, un sommet s
Ensure: Un parcours en largeur L d'origine s, les valeurs $dist_s(u), u \in V$
  for all $u \in V$ **do**
    $dist_s(u) := +\infty$
  end for
  $L := ()$, Enfiler (F,s), $dist_s(s) := 0$
  while not FileVide(F) **do**
    u :=Défiler(F), $L := L + (u)$
    for all $\{u, v\} \in E$ **do**
      if $dist_s(v) = +\infty$ **then**
        Enfiler(F,v), $dist_s(v) = dist_s(u) + 1$
      end if
    end for
  end while

Question 1
Appliquer cet algorithme au graphe G_1 en partant du sommet $s = 4$. Préciser, à chaque itération, le sommet u retiré à F, le sous-parcours L, la valeur de la file F, la valeur des $dist_s(v)$ pour $v \in V_1$. Les valeurs de L, de F et de $dist_s$ sont celles obtenues à chaque itération en fin du corps de boucle.

Exercice 5 – Complexité du calcul d'un parcours en largeur

On considère un graphe non orienté connexe $G = (V, E)$ ayant n sommets et m arêtes. Le but de cet exercice est d'évaluer la complexité du calcul d'un parcours en largeur. On suppose que :
— L et F sont stockés dans des listes circulaires doublement chaînées ;
— les distances $dist_s$ sont stockées dans un tableau $D[1 \dots n]$;
— le graphe non orienté est représenté par une matrice sommet-sommet, une matrice sommet-arête ou des listes d'adjacences.

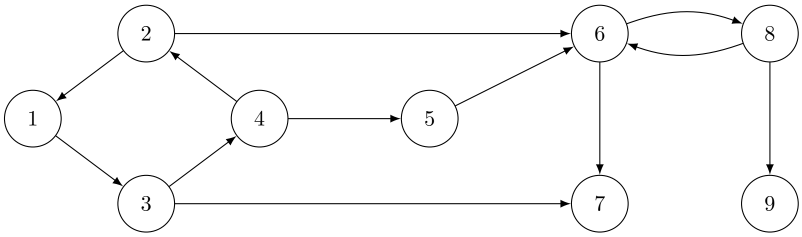
Question 1
Pour tout sommet $u \in V$, on note $cv(u)$ le coût du calcul des sommets adjacents à u.

1. Rappeler sans explication l'ordre de grandeur de $cv(u)$ en fonction de la représentation de G ;
2. Donnez en fonction de $cv(u)$ l'ordre de grandeur de la boucle interne de l'algorithme.

Question 2
En déduire la complexité de l'algorithme de calcul d'un parcours en largeur en fonction de cv, puis pour chacune des représentations des graphes non orientés.

Exercice 6 – Parcours en largeur des graphes orientés

Question 1
On considère le graphe orienté $G_3 = (V_3, A_3)$:



Pour chaque racine de G_3, donner un parcours en largeur de G_3.

Question 2

Adapter l'algorithme de parcours en largeur au cas des graphes orientés. L'appliquer au graphe G_3, en partant de l'une des racines.