

TD n 7

Parcours en largeur

On rappelle l'algorithme de **parcours en largeur** (ou BFS comme *Breadth-first search*) en partant de r . L est une file (une liste chaînée...)

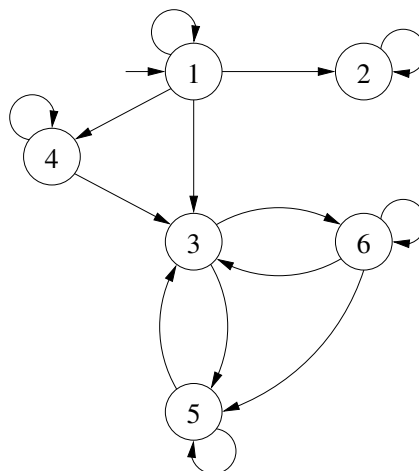
début

```

    couleur[r] = gris
    Liste L = (r) /* la liste ne contient que r */
    tant que L n'est pas vide faire
        x = ExtraireTeteDeListe(L)
        pour tout voisin y de x faire
            si couleur[y] = blanc alors
                InsérerEnQueueDeListe(L, y)
                couleur[y] = gris
                pere[y] = x
        couleur[x] = noir

```

Exercice 1 Appliquer au graphe ci-dessous l'algorithme de parcours en largeur en partant de 1 (indiqué par une flèche entrante). L'arbre de parcours en largeur résultant sera présenté par un schéma dans lequel les sommets de profondeur égale seront mis à la même hauteur, le sommet origine étant mis en haut.



Exercice 2 Quand la file L devient vide, décrire ce que sont les sommets blancs, gris et noirs.

Exercice 3 Une composante connexe d'un graphe **non-orienté** est une classe d'équivalence de la relation "il y a un chemin de x à y ". Écrire un algorithme qui affiche les composantes connexes d'un graphe.

Exercice 4 On fait un parcours en largeur d'un graphe **orienté**. Un arc $x \rightarrow y$ du graphe est dit

- Une boucle si $x = y$
- De parcours si $pere[y] = x$
- En avant si y descend de x dans l'arbre
- En arrière si x descend de y dans l'arbre
- Transverse sinon

Quels sont les types d'arcs possibles et ceux impossibles ?

Exercice 5 Et maintenant on parcourt en largeur un graphe **non orienté**. Une arête xy est dite

- De parcours si $pere[y] = x$ ou $pere[x] = y$
- Parallèle si y descend de x ou x descend de y dans l'arbre
- Transverse sinon

Quels sont les types d'arêtes possibles et ceux impossibles ?

Exercice 6 Implémenter l'algorithme selon que le graphe est codé en

1. listes d'adjacence
2. matrice d'adjacence

(ce qui change est la façon d'obtenir les voisins de x). Analyser, à chaque fois, la complexité.

Exercice 7 Modifier l'algorithme pour qu'il calcule les distances et affiche les sommets classés par distance croissante de r . Exemple d'affichage :

À distance 0 du sommet 0 : sommets 0
À distance 1 du sommet 0 : sommets 2 4 5
À distance 2 du sommet 0 : sommets 8 1
À distance 3 du sommet 0 : sommets 6 7

Exercice 8 Proposer un algorithme renvoyant, s'il existe, un circuit de longueur minimale passant par un sommet s donné dans un graphe G . Prouver sa correction.