TP : Parcours de graphes

I) Notions élémentaires

Exercice 1 **1)** `Lex1 = [[1, 2, 3, 4], [0, 2, 3], [0, 1, 3], [0, 1, 2], [0, 5], [4], [7], [6]]`

2) Le sommet `2` a pour degré $d(2) = 3$ (arêtes vers `0`, `1` et `3`), et le sommet `5` a pour degré $d(5) = 1$ (arête vers `4`).

3) Le chemin `[3, 0, 4, 5]` relie `3` à `5`, et est de longueur 3 (3 arêtes).
Le chemin `[3, 2, 1, 0, 4, 5]` convenait aussi, et est de longueur 5.

4) Cycle de longueur 4 : `[3, 2, 1, 0, 3]`. De longueur 3 : `[3, 0, 1, 3]` (il y en a d'autres).

5) Composantes connexes : le sous-graphe constitué des sommets `[6, 7]`, et celui des sommets `[0, 1, 2, 3, 4, 5]`.

Exercice 2 **1)** `L=[[1, 2], [2, 3], [1, 3], [4], [0]]`

2) $d_-(2) = 2$ et $d_+(2) = 2$. $d_-(3) = 2$ et $d_+(3) = 1$.

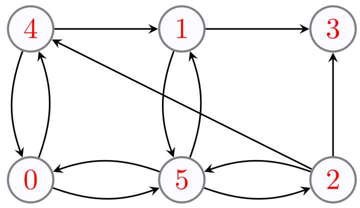
3) `[0, 2, 3, 4]` (longueur 3) ou `[0, 1, 3, 4]` (idem).

4) Cycle de longueur 2 : `[2, 1, 2]`. Cycle de longueur 4 : `[0, 1, 3, 4, 0]`.

Exercice 3

1) Le graphe est orienté : Il n'y a pas d'arête depuis le sommet 3, alors qu'il y a des arêtes vers 3.

2)



3) $M = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}$

II) Parcours en largeur

Vous trouverez ici :

`http://mpechaud.fr/scripts/parcours/index.html`

une très belle animation de plusieurs algorithmes de parcours de graphe.

Exercice 4 **8)** Posons $n = |S|$. Il y $3n$ $(+2)$ affectations lors de l'initialisation.
La boucle **while** dure autant qu'il y a d'éléments qui passent dans `f`.
Or chaque sommet passe dans `f` une fois et une seule (passage de blanc à gris).
Donc il y a exactement n passages dans la boucle **while**.
Plus généralement (boucles **while** et **for**), il y a autant d'affectations que

- de passage de blanc à gris (`c[a]`, `p[a]`, `d[a]`), donc n
- de passage de gris à noir (c[u]='n') donc n

Ainsi le nombre d'affectation est en O(|S|)
Pour la taille des boucles, les boucles **for** parcourent les arêtes partant de `u`, et `u` parcourt l'ensemble des sommets une fois et une seule. On parcourt donc l'ensemble des arêtes une fois et une seule. La taille cumulée des boucles **for** est donc $|A|$.
La complexité est donc en $O(|S| + |A|)$.

III) Parcours en profondeur

Exercice 5 **4)** Toujours avec $n = |S|$. Il y a toujours $O(n)$ affectations lors de l'initialisation.

La coloration blanc $\longrightarrow$ gris assure que la fonction `visiter_sommet` est appellée une fois et une seule sur chaque sommet, elle est donc appellée exactement $n = |S|$ fois.

Dedans il y a au moins 2 affectations (`c[u]` à gris, puis noir). Les boucles **for** dans leur ensemble parcourent chaque arête une fois et une seule : il y a donc $|A|$ passages dans les boucles **for**.

En conclusion, l'algorithme est de complexité $O(|S| + |A|)$