THEORIE DES GRAPHES

4) IMPLANTATION D'UN GRAPHE

Parmi les différentes implantations possibles d'un graphe, on peut en distinguer deux qui interviennent dans la plupart des problèmes pratiques et théoriques, à savoir, la matrice d'adjacence et la liste des successeurs.

4.1. Matrice d'adjacence :

Dans le cas où l'ensemble des sommets du graphe G est fixé à l'avance, on peut représenter G par sa matrice d'adjacence. Le type correspondant est alors le suivant.

type GRAPHE = tableau [1 n, 1 n] de booléens (ou d'entier selon la convenance) ; où n dénote l'ordre de G.

Soit A une variable de type GRAPHE représentant un graphe G donne. On a donc

A[i, j] = vrai (ou 1) si et seulement si il existe un arc i—>j dans G, et faux (ou 0) sinon.

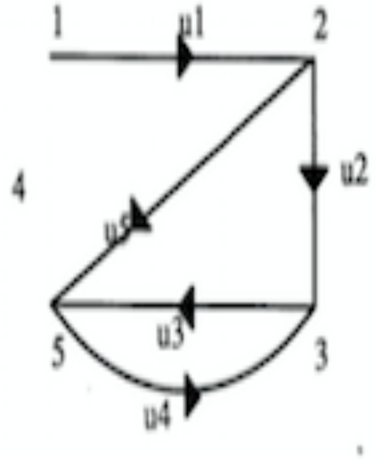
Pour les algorithmes sur les graphes, on évalue la complexité en place et en temps en fonction de deux paramètres, le nombre n de sommets et le nombre m d'arcs ou d'arêtes.

Avec la représentation par matrice d'adjacence, la complexité en place est en $\Theta(n^2)$ et ce, quelque soit le nombre d'arcs ou d'arêtes du graphe. Aussi, si le graphe comporte peu d'arcs ou d'arêtes, cette représentation devient inadéquate. Par ailleurs, avec une telle représentation, la complexité en temps de la recherche des successeurs (resp. des prédécesseurs) d'un sommet est en $\Theta(n)$ (car les successeurs (resp. les prédécesseurs) d'un sommet i s'obtiennent en examinant la ième ligne (resp. ième colonne)). Notons que la consultation complète de la matrice d'adjacence requiert un temps d'ordre n^2.

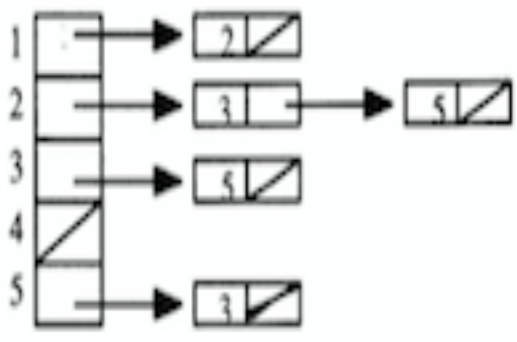
4.2. - Liste des successeurs :

A chaque sommet, on associe la liste de ses successeurs dans un certain ordre. Afin de pouvoir accéder à ces listes, on crée un tableau qui contient, pour chaque sommet, un pointeur vers le début de sa liste des successeurs.

Exemple



Le graphe G.



La liste des successeurs du graphe G.

Le type correspondant est le suivant.

type adr = ^doublet

doublet = enregistrement

no : entier

suiv : adr

Fin ;

GRAPHE = tableau [1 … n] d'adr

Soit A une variable de type GRAPHE représentant un graphe donné.

A[i] est l'adresse du début de la liste des successeurs du sommet i (A[i] contient un pointeur sur une liste chaînée de tous les successeurs du sommet i).

A[i]^.no est le premier successeur du sommet i.

Avec la représentation par liste de successeurs, la complexité en place est en $\Theta(n + m)$ pour un graphe orienté (il y a autant de doublets que d'arcs) et en $\Theta(n + 2*m)$ pour un graphe non orienté.

Aussi, si le graphe comporte peu d'arcs ou d'arêtes, cette représentation est alors préférable à la représentation par matrice d'adjacence.

Par contre, tel n'est plus le cas pour un graphe dense.

Par ailleurs, avec une telle représentation, la complexité en temps de la recherche des successeurs d'un sommet i est en $\Theta(d^+_G(i))$ (car l'examen des successeurs d'un sommet i se ramène à un parcours de la liste A[i].

Aussi, l'examen de tous les sommets d'un graphe peut se faire en un temps d'ordre m.

Par contre, cette représentation n'est plus adaptée au traitement des prédécesseurs d'un sommet i car, pour ce faire, il faut parcourir toutes les listes des successeurs et comparer les numéros des sommets entre eux, soit m parcours de liens pointeurs et m comparaisons de numéros de sommets.