Correction du TD n°1

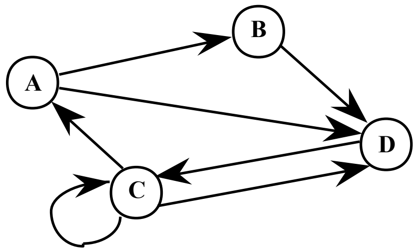
Exercice 1 :

Rappel :

- un chemin est une suite de sommets [x0 ..xn] tels que les arcs (xi,xi+1) appartiennent au graphe
- un chemin élémentaire ne passe pas 2 fois par le même sommet
- un chemin simple ne passe pas 2 fois par le même arc

Degrés, etc.…

Soit le graphe



1) Enumérer: U(A), U(B), $U^-(C)$,$U^+(D)$.

U(A) = {B, D, C} (l'ensemble des sommets adjacents à A);

U(B) = {D,A}, $U^-(C)$ = {C,D},$U^+(D)$={C}.

2) $d^+(C) = 3$, $d^-(C) = 2$, $d(C) = 4$.

3) Rapporter un chemin simple mais pas élémentaire.

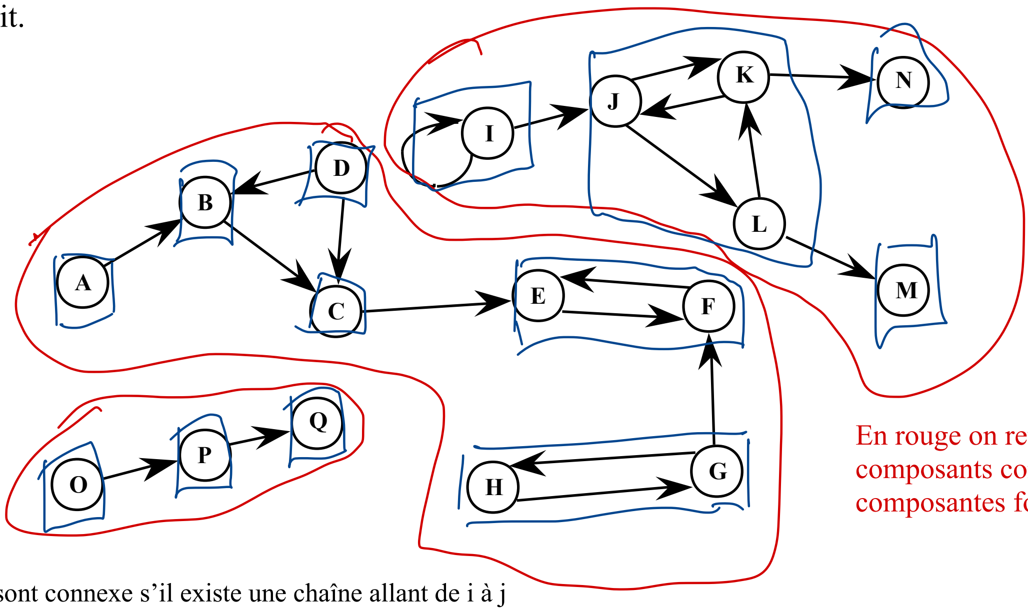
[A, B, D, C, A, D]

4) Rapporter un circuit Hamiltonien.

[A, B, D, C, A]

Exercice 2 :

On considère le graphe G=(X,U) ci-dessous. Rapporter des chaînes de B à G de longueur 7, 8, 9, 10 et 11. Déterminer ses composantes connexes, ses composantes fortement connexes ainsi que le graphe réduit.



En rouge on retrouvera les composants connexes et en bleu les composantes fortement connexes.

Rappel :

- i et j sont connexe s'il existe une chaîne allant de i à j
- i et j sont fortement connexe s'il existe un circuit allant de i à j
- graphe réduit = graphe des composantes fortement connexes

Relation d'équivalence…

origine Terminaison

Chaîne de B à G de longueur 7 : (DB) (DC) (CE) (EF) (FE) (EF) (GF).

Chaîne de B à G de longueur 8 : (BC) (CE) (EF) (FE) (EF) (GF) (GH) (HG).

Exercice 3 :

Montrer le lemme de KOENIG : on peut toujours extraire d'un chemin allant de i à j un chemin élémentaire allant de i à j. Généraliser à une chaîne.

Preuve : Soit μ une chaîne quelconque allant de i à j et supposons μ non élémentaire alors il existe p tel que: μ = [i...p ...p...j]. Notons $\mu 1$ = [i ...p], le chemin de i au premier p dans le chemin μ, et μ_2 = [p ...j] le chemin du dernier p au j. clairement $\mu' = \mu 1 + \mu 2$ donne un chemin allant de i à j ne passant qu'une fois par p. Si μ' n'est pas élémentaire, on trouvera un sommet q et on répète la procédure sur le chemin μ' et le sommet q. On obtiendra alors un chemin μ'' allant de i à j ne passant qu'une fois par p et q. De cette façon on extraira de μ des chemins de i à j ne passant au plus qu'une fois par chacun des sommets du graphe, donc un chemin élémentaire.

Exercice 4 :

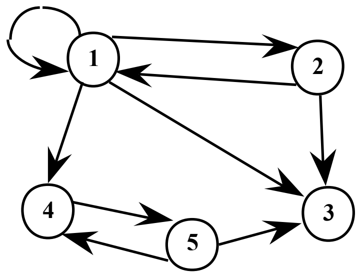
Quelles méthodes proposez-vous pour coder un graphe en machine ?

Matrice d'incidence, d'adjacence et tableaux des successeurs.

Exercice 5

On propose deux méthodes pour coder un graphe G=(X,U) en machine : la matrice d'adjacence A et la file des successeurs de tableaux ALPHA1, ALPHA2 et BETA. La matrice associée est définie par $A=(a_{ij})$ avec $a_{ij}=1$ si l'arc (i,j) appartient à U et $a_{ij}=0$, sinon. La file des successeurs BETA est le tableau des successeurs, les successeurs du sommet I se trouvant entre les adresses ALPHA1(I) et ALPHA2(I) dans le tableau BETA, (en cours on avait un seul tableau ALPHA).

Rapporter les tableaux A, ALPHA1, ALPHA2 et BETA pour le graphe ci-dessous.



Ecrire un algorithme permettant de passer de la file des successeurs à la matrice associée. Quelle est la complexité de cet algorithme?

Ecrire un algorithme permettant de passer de la matrice associée à la file des successeurs. Quelle est la complexité de cet algorithme ?

Remarque:

On a en fait deux matrices associées car les valeurs peuvent être entières (quand on donne les valuations associées aux arcs) ou booléennes (quand c'est l'existence des arcs).

Exercice 5

A=

	1	2	3	4	5
1	1	1	1	1	0
2	1	0	1	0	0
3	0	0	0	0	0
4	0	0	0	0	1
5	0	0	1	1	0

BETA =

1	2	3	4	1	3	5	3	4
1	2	3	4	5	6	7	8	9
1				2		4	5	

ALPHA1	1	5	0	7	8
ALPHA2	4	6	0	7	9

et

ALPHA	1	5	7	7	8	10

```
{ file des successeurs vers matrice d'adjacence }
for i := 1 to n do
for j := 1 to n do
A[i,j] := 0;
for i := 1 to n do
begin
j1 := ALPHA1[i];
j2 := ALPHA2[i];
if ( j1 <> 0 ) then
     begin
          for j := j1 to j2 do
               A[i, BETA[j]] := 1;
     end;
end;
```

$O(n^2)$

$d^+(i)$

$O\left(\sum_{i=1}^{n} d^+(i)\right) = O(m)$

complexité = $O(n^2) + O(m) \sim O(n^2)$.

Explication:
Au début, on initialise la matrice A avec des valeurs nulles, donc on part d'un graphe vide qu'on remplira graduellement.
j1 et j2 sont deux variables où on stockera l'indice du premier successeur et dernier successeur du sommet i courant dans la file beta.
On commence dans la premiere boucle d'examiner les sommets un par un. On remarque que pour chaque sommet i on a access directement aux successeurs grace aux indice j1 et j2 et puisque BETA[j]] pour j entre j1 et j2 donne un successsuer de i, on marquera A[i, BETA[j]] := 1.
Concernant la complexité, on remarque que la premiere boucle imbriquée donne une complexité de O(n2) et la deuxième O(m).

```
{ matrice d'adjacence vers file des successeurs }

indi := 1;
for i :=1 to n do
begin
    j1 := 0;
    j2 := 0;
    for j := 1to n do
        if A[i,j] = 1 then
            begin
                if j1 = 0 then j1 := indi;
                BETA[indi] := j;
                j2 := indi;
                indi := indi + 1;
            end;
    ALPHA1[i] := j1;
    ALPHA2[i] := j2;
end;
Complexité = O(n²).
```

$O(1)$

$O(n)$

$O(n^2)$

Explication:
la variable indi sert à indiquer la position dans le file beta du prochain successeur à rajouter.
Dans la boucle principale, on examine les sommets un par un. Lors de l'examen d'un sommet i, on initialise j1 et j2 = 0 car on ne sait pas si le sommet i a des successeurs ou pas. Ensuite on parcours la ligne i de la matrice A et dès qu'on trouve un premier A[i,j] = 1, (donc j est le premier successeur de i) on met j1 egale à la valeur courante de indi et on stocke dans beta le prochain element (qui est le successeur j de i). Chaque fois qu'on trouve un successeur de i on suppose qu'il peut etre le dernier et on affecte à j2 la valeur de indi courant et on le rajoute à beta. A la fin du parcours de la ligne i, j1 et j2 contient l'indice du premier et dernier successeur du sommet i dans la file beta.

Exercice 6 :

Donner le nombre d'arêtes d'un graphe non orienté complet de n sommets.

Pour le graphe non orienté sans boucles on a $C_n^2 = n*(n-1)/2$, car chaque arête est vu comme une paire de sommets, il faut alors trouver le nombre de différentes de paires de sommets dans le graphe, ou le nombre de sous-ensembles de deux éléments dans un ensemble de n éléments (C_n^2).

Exercice 7 :

Expliquer pourquoi si G=(X,E) est un graphe non orienté sans boucle, la somme des degrés est égale à deux fois le nombre d'arêtes du graphe.

Réponse : parce que l'on compte 2 fois les arêtes. En fait, chaque arête est comptée dans le calcul des degrés de chacun de ses extrémités.

Complément : Ci-dessous la preuve du lemme de Koenig par recurrence sur la longueur des chemins.

Montrer le lemme de KOENIG : on peut toujours extraire d'un chemin allant de i à j ($i \neq j$) un chemin élémentaire allant de i à j. Généraliser au cas i=j, puis à une chaine et un cycle.

Demonstration par recurrence sur le nombre K - longueur du chemin.

La propriété : "On peut toujours extraire d'un chemin de i à j ($i \neq j$) un chemin élémentaire de i à j"

Pour $K=1$: le chemin est réduit à l'arc (i,j) => la propriété est vérifiée.

Supposons que la propriété est vraie pour long. $\leq K$ et démontrons qu'elle reste vérifiée pour tout chemin de longueur $K+1$.

Soit $\mu = [i \dots j]$, un chemin de i à j de long. $K+1$.

Il y a 2 possibilités : (1) μ est élémentaire => propriété vérifiée

(2) μ n'est pas élémentaire : $\exists\, p$ tel que $\mu = [\underbrace{i \dots p}_{\mu_1} \overbrace{\dots p}^{\mu_2} \underbrace{\dots j}_{\mu_3}]$. => $long(\mu_2) \geq 1$, $\mu' = \mu_1 + \mu_3$ et $long(\mu') \leq K$ => On peut appliquer l'hypothèse de recurrence sur μ' => la propriété est vérifiée.