

Théorie des graphes

1– Éléments de théorie des graphes

La théorie des graphes concerne de nombreux domaines d'application. Les objectifs sont distincts mais le langage utilisé est le même.

Les graphes sont utilisés pour modéliser des ensembles structurés complexes. Leurs applications sont nombreuses, par exemple la modélisation dans le temps, modélisation des réseaux etc ...

La théorie des graphes est née en 1736 quand Euler démontre qu'il est impossible de traverser chacun des 7 ponts de la ville de Königsberg, une seule fois et en revenant au point de départ.

De manière générale un graphe permet de représenter simplement la structure, les connexions, les chemins possibles d'un ensemble complexe, comprenant un grand nombre de situations en exprimant les relations de dépendances entre les éléments.

1.1- Définition et concept de base

1.1.1 Concepts orientés

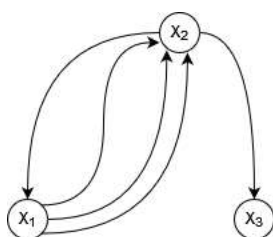
Dans beaucoup d'applications les relations entre les éléments d'un ensemble sont orientées. Un élément X peut être en relation avec un élément Y , sans que Y ne soit nécessairement en relation avec X .

Un graphe $G=(X, U)$ est déterminé par :

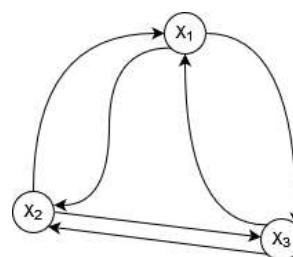
- Un ensemble fini X ; $X=\{x_1, x_2, \dots, x_n\}$ dont les éléments sont appelés **sommet** ou **nœud**. Si X possède n éléments, le graphe sera dit d'**ordre n** .
- Un ensemble fini $U=\{u_1, u_2, \dots, u_n\}$ dont les éléments sont des couples ordonnés de sommets appelés **arcs**. U est donc une famille de produit cartésien.

$$X \times X = \{(a, b) \mid a \in X, b \in X\}$$
 Pour un arc $u=(n_i, n_j)$, n_i est l'extrémité initiale et n_j celui final.
- Un arc (n_i, n_i) est appelé une boucle.
- Un **p-graphe** est un graphe dans lequel il n'existe jamais plus de p -arcs de la forme (x_i, x_j) , entre deux sommets quelconques, c'est à dire le nombre maximum d'arcs de même sens liant deux sommets distincts x_i, x_j .

Exemple :



$p=3$



$p=1$

Application multivoques :

- x_j est successeur de x_i
 $(x_i, x_j) \in U \quad \Gamma(x_i)$
- x_j est prédécesseur de x_i
 $(x_j, x_i) \in U \quad \Gamma^{-1}(x_i)$
- L'application Γ qui à tout éléments de X fait correspondre une partie de X , elle est appelé application multivoques (→ plusieurs sens ?)

Pour un 1-graphe, j'ai peut être parfaitement déterminée par (X, Γ) qui est 1 notation à la base d'une représentation informatique très utilisé : les listes adjacences.

Exemple :

$$X = \{x_1, x_2, x_3, x_4, x_5\}$$

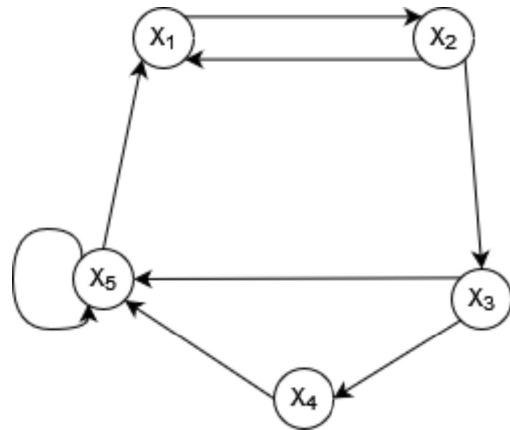
$$\Gamma(x_1) = \{x_2\}$$

$$\Gamma(x_2) = \{x_1, x_3\}$$

$$\Gamma(x_3) = \{x_4, x_5\}$$

$$\Gamma(x_4) = \{x_5\}$$

$$\Gamma(x_5) = \{x_1, x_5\}$$



Remarque :

On peut également définir une fonction w telle que

$w^+(x_i)$ **représente un ensemble des arcs sortant** de x_i et réciproquement

$w^-(x_i)$, **l'ensemble des arcs entrant**.

1.1.2- Concepts non orienté

Lors de l'étude de certaine propriété il arrive que l'orientation des arcs ne joue aucun rôle. On s'intéresse à l'existence d'arcs entre deux sommet sans préciser leur ordre. Un arcs sans orientation est appelé arrête. Pour une arrête (x_i, x_j) , on dit que u est incidente aux deux sommets.

Remarque :

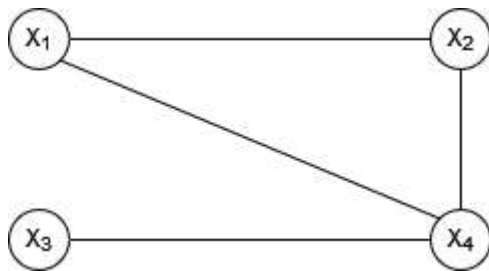
Dans le cas non orienté au lieu de $G=(X, U)$ et $u=(x_i, x_j)$, on préfère souvent :

$$G=(X, E) \text{ et } e=[x_i, x_j]$$

Un multigraphe $G=(X, E)$ est un graphe dans lequel il peut exister plusieurs arrête entre deux sommets distinct.

Exemple :

Graphe non orienté



1.1.3 Principales définition

Adjacence :

Deux sommets sont adjacents ou voisins s'ils sont joints par un arc.

Deux arcs sont adjacents s'ils ont au moins une extrémité commune.

Degré :

Le **demi degré extérieur** de x_i , $d^+(x_i)$ est le nombre d'arcs ayant x_i comme extrémité initiale. $d^+(x_i) = \text{card}(w^+(x_i))$

Le **demi degré intérieur** de x_i , $d^-(x_i)$ est le nombre d'arcs ayant x_i comme extrémité finale.

$$d^-(x_i) = \text{card}(w^-(x_i))$$

Le **degré** de x_i est $d(x_i) = d^+(x_i) + d^-(x_i)$.

Graphes complémentaires :

$$G = (U, X) \text{ et } \overline{G} = (X, \overline{U})$$

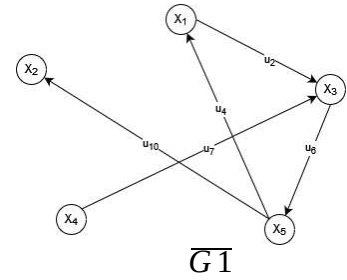
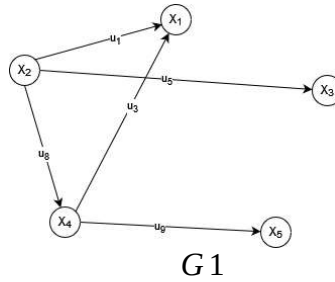
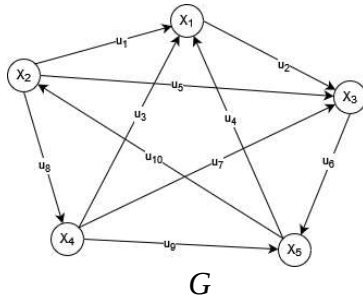
$$(x_i, x_j) \in U \Rightarrow (x_i, x_j) \notin \overline{U} \text{ et}$$

$$(x_i, x_j) \notin U \Rightarrow (x_i, x_j) \in \overline{U}$$

\overline{G} Est le graphe complémentaire de G .

Deux graphes sont complémentaires s'il n'y a aucune arête commune et si leur réunion donne un graphe complet.

Exemple :



Graphes partiel :

$$G = (X, U) \text{ et } U_p \subset U$$

$G_p = (X, U_p)$ est un graphe partiel on peut obtenir des sommets isolés.

Sous-graphes :

$$G = (X, U) \text{ et soit}$$

$X_s \subset X$ un sous ensemble de X .

$G_s = (X_s, V)$ est un sous graphes de G où V est l'ensemble des arcs de U dont les extrémités sont dans X_s .

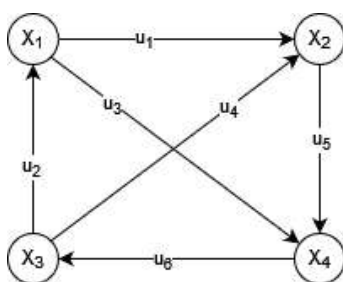
Sous-graphes Partiel :

$G = (X, U)$ soient $A \subset X$ et $V_A \subset U$, le sous-graphe partiel engendré par $V \cap U_p$ est

$$G_{sgp} = (A, V_A).$$

Cette définition combine les deux précédentes.

Exemple :



$$G = (X, U)$$

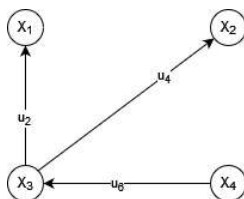
$$X = \{x_1, x_2, x_3, x_4\}$$

$$U = \{u_1, u_2, u_3, u_4, u_5, u_6\}$$

Graphe partiel :

$$X = \{x_1, x_2, x_3, x_4\}$$

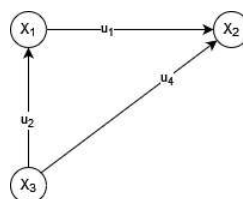
$$U_p = \{u_2, u_4, u_6\}$$



Sous-graphe :

$$X_s = \{x_1, x_2, x_3\}$$

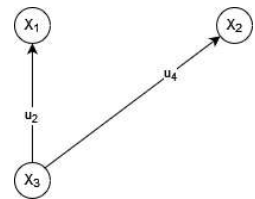
$$U = \{u_1, u_2, u_4\}$$



Sous-graphe partiel :

$$X = X_s = \{x_1, x_2, x_3\}$$

$$V_A = V \cap U_p = \{u_2, u_4\}$$



Graphes réflexifs :

$$\forall x_i \in X, (x_i, x_i) \in U$$

Autrement dit quelque soit le sommet du graphes il va y avoir une boucle.

Graphes irreflexif

$$\forall x_i \in X, (x_i, x_i) \notin U$$

Autrement dit quelque soit le sommet du graphes, il ne doit pas y avoir de boucles.

Graphes symétrique

$$\forall x_i \in X, \forall x_j \in X, (x_i, x_j) \in U \Rightarrow (x_j, x_i) \in U$$

Autrement dit quelque soit deux sommet quelconque distinct, il n'y a aucun ou deux arcs de sens opposé.

Graphes asymétrique

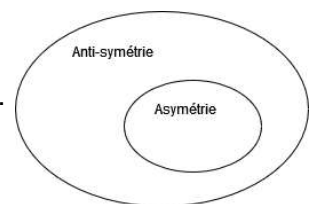
$$\forall x_i \in X, \forall x_j \in X, (x_i, x_j) \in U \Rightarrow (x_j, x_i) \notin U$$

Si G est asymétrique alors G est irreflexif. C'est-à-dire qu'il n'y a pas de boucle.

Graphes anti symétrique

$$\forall x_i \in X, \forall x_j \in X, (x_i, x_j) \in U, (x_j, x_i) \in U \Rightarrow x_i = x_j$$

Cette définition prend en compte l'existence de boucle dans le graphes. L'anti-symétrie inclue l'asymétrie. L'asymétrie \Rightarrow anti-symétrique.



Graphes transitif

$$\forall x_i \in X, \forall x_j \in X, \forall x_k \in X, (x_i, x_j) \in U, (x_j, x_k) \in U \Rightarrow (x_i, x_k) \in U$$

Graphes complet

$$\forall x_i \in X, \forall x_j \in X, x_i \neq x_j, (x_i, x_j) \notin U \Rightarrow (x_j, x_i) \in U$$

Autrement dit un graphes est complet lorsque entre **deux sommet quelconque distinct il y a au moins un arc.**

Graphes simple

Un graphes est dit simple si tout les arcs sont distinct et s'il n'y a pas de boucles.

Clique

Une clique de $G=(X, U)$ est un sous graphe complet de G.

C'est à dire un sous graphes $\tilde{G}=(\tilde{X}, E)$, $\tilde{X} \subset X$ tel que tout couple de sommet distinct de \tilde{X} est relié par une arrête de \tilde{E} .

Un graphes complet et asymétrique s'appelle un tournoi.

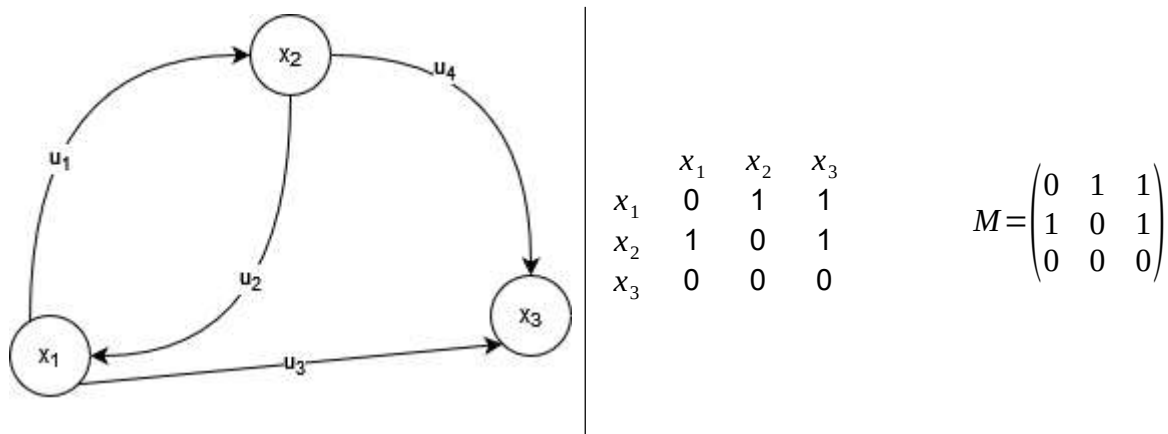
1.2 Représentation d'un graphe

Il existe un certain nombre de représentation pour décrire un graphes. Celle ci ne sont pas équivalentes aux niveau efficacité algorithmique. On distingue principalement la représentation par matrice d'adjacence, par matrice d'incidence sommets-arcs ou (sommets-arrêtes) et par liste d'adjacence.

1.2.1 Matrice d'adjacence

On considère un 1-graphe, la matrice d'adjacence fait correspondre les sommet origine des arcs (placé en ligne dans la matrice) aux sommets destination (placé en colonne). Dans le formalise matrice booléenne l'existence d'un arcs (x_i, x_j) ce traduit par la présence d'un 1 à l'intersection de la ligne x_i et de la colonne x_j . L'absence d'arcs ce traduit par un 0.

Exemple :



La place mémoire utilisé pour un graphe d'ordre n est n^2

1.2.2 Matrice d'incidence sommets-arcs

Pour cette représentation les sommets sont placé en ligne et les arcs en colonne. Si $u = (x_i, x_j) \in U$, on trouve dans la colonne u : $a_{iu} = 1$ et $a_{ju} = -1$ tous les autres termes de la colonne sont nul.

Exemple :

	u_1	u_2	u_3	u_4
x_1	1	-1	1	0
x_2	-1	1	0	1
x_3	0	0	-1	-1

$$A = \begin{pmatrix} 1 & -1 & 1 & 0 \\ -1 & 1 & 0 & 1 \\ 0 & 0 & -1 & -1 \end{pmatrix}$$

La place en mémoire utilisé est $n * m$ où n est l'ordre du graphe et m le nombre d'arcs.
La somme de chaque colonne est 0.

1.2.3 Listes d'adjacence

Pour un 1-graphe l'avantage de la représentation par liste d'adjacence (**grâce à l'application multivoque Γ**) par rapport à celle de la matrice d'adjacence est le gain obtenu en place mémoire. Ce type de représentation est mieux adapté pour une implémentation.

Le but est de représenter chaque arcs par son extrémité finale, l'extrémité initiale étant définis implicitement. Tous les arcs émanant d'un même sommet sont liés entre eux dans une liste. À chaque arcs sont donc associés le sommet destination et le pointeur au prochain sommet dans la liste.

La représentation nécessite la création de deux tableaux. Un premier nommé L_p (liste pointeur) de dimension $n+1$ et un second L_s (liste successeur) de dimension m pour le cas orienté et $2m$ dans le cas non orienté.

Pour chaque sommet i , la liste des successeurs de i est contenu dans le tableau L_s à partir de la case $L_p(i)$.

L'ensemble des informations relatives au sommet i est contenu entre les cases $L_p(i)$ et $L_p(i+1)-1$

Cas orienté :

$$\text{card}(w^+(i)) = L_p(i+1) - L_p(i)$$

Cas non orienté :

$$\text{card}(w(i)) = L_p(i+1) - L_p(i)$$

$$L_p(i) = \sum_{j=1}^{i-1} \text{card}(w^+(j)) + 1$$

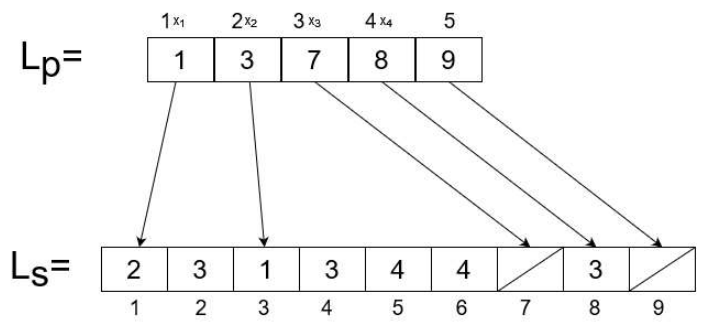
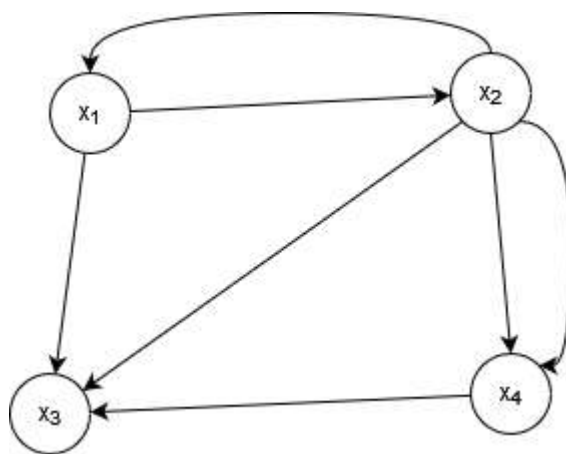
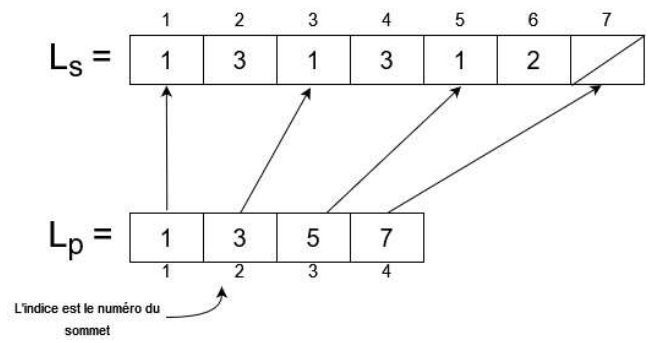
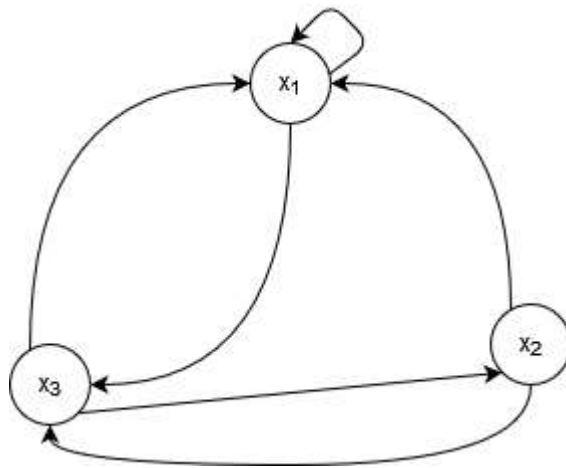
Cette représentation revient à décrire le graphe par son application multivoque $i \rightarrow \Gamma(i)$.

Méthode :

1. On construit L_s par $\Gamma(1), \Gamma(2), \dots, \Gamma(n)$
2. On construit L_p qui donne pour tout sommet, l'indice dans L_s où commence ces successeurs.
3. Pour un sommet $i \leq \text{premier suivant est dans } L_s(L_p(i))$.
Pour le deuxième $L_s(L_p(i+1))$
4. Si un sommet i n'a pas de successeur on pose $L_p(i) = L_p(i+1)$ soit une liste vide coincée entre les successeurs de $i-1$ et $i+1$.

Pour éviter de tester les cas particuliers, $i=n$, le sommet $i+1$ n'existant pas, on se conforme par convention la dernière liste en posant $L_p(n+1) = m+1$.

Exemple :



Il est conseillé de mettre la liste L_p au dessus de la liste L_s .

1.3 Coloration d'un graphe

1.3.1 Coloration des sommets

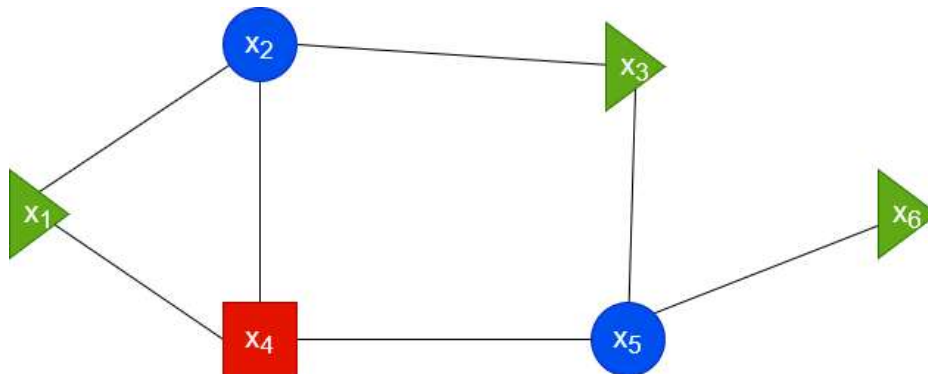
Définitions :

Soit $G=(X, U)$ un sous ensemble $S \subset U$ est un ensemble stable s'il ne comprend que des sommets non adjacents deux à deux.

$$\forall x_i \in S, \forall x_j \in S \Rightarrow (x_i, x_j) \notin U$$

Comme tous sous-ensemble d'un ensemble stable est un ensemble stable, il est naturel de rechercher le cardinal maximum d'un ensemble stable. Ce nombre est noté $\alpha(G)$ et est appelé **nombre de stabilité**.

Exemple :



Le nombre chromatique $\gamma(G)$ est défini comme le nombre de couleur minimum distinct nécessaire à la coloration des sommets de G .

Un graphe G tel que $\gamma(G) \leq k$ qui est colorable en k -couleur est dit k -chromatique.

Une k -coloration des sommets, est une partition (S_1, S_2, \dots, S_k) de l'ensemble en k -ensemble stable.

Borne inférieure du nombre chromatique

Puis que $\alpha(G)$ est le nombre maximum des sommets des ensembles stables déduits de G on a $\alpha(G) * \gamma(G) \geq n(G)$.

$$S_1 = \{x_1, x_3, x_6\}$$

$$S_2 = \{x_2, x_5\}$$

$$S_3 = \{x_4\}$$

$$\alpha(G) = \max(\text{card}(S_1), \text{card}(S_2), \text{card}(S_3))$$

$$\alpha(G) = \max(3, 2, 1)$$

$$\alpha(G) = 3$$

$$n(G) = 6$$

$$\gamma(G) \geq \frac{6}{3} = 2 \Rightarrow \gamma(G) \geq 2$$

Borne Supérieure du nombre chromatique

Une borne supérieure est donnée par $\deg + 1$
avec $\deg = \text{plus grand degré des sommets du graphe}$.

$$\frac{n(G)}{\alpha(G)} \leq \gamma(G) \leq \deg + 1$$

$$2 \leq \gamma(G) \leq 4$$

Algorithme de coloriage d'un graphe (pour les sommets)

1. On range les sommets dans un tableau un ordre de degré décroissant.
2. On attribue la couleur c_1 à x_1 et aux sommets suivants de la liste qui n'est pas adjacent à x_1 et ainsi de suite avec les sommets de la liste qui ne sont pas adjacents aux sommets déjà coloriés.
3. On attribue ensuite la couleur c_2 au premier sommet non colorié ainsi qu'aux sommets suivants qui ne sont pas adjacents aux sommets coloriés par la couleur c_2 .
4. On continue le processus jusqu'à épuisement des sommets dans la liste.

Sommets	Degré	Couleur
x_2	3	c_1
x_4	3	c_2
x_5	3	c_1
x_1	2	c_3
x_3	2	c_2
x_6	1	c_2

1.4 Connexité dans les graphes

1.4.1 Chaîne-cycle

Chaîne :

Une chaîne de longueur q (de cardinalité q) est une séquence de q arcs $L = \{u_1, u_2, \dots, u_q\}$ tel que chaque arc u_r de la séquence ($2 \leq r \leq q-1$) ait une extrémité commune avec u_{r-1} , ($u_{r+1} \neq u_r$) et l'autre extrémité avec u_{r+1} , ($u_{r+1} \neq u_r$).

L'extrémité x_i de u_1 non adjacent à u_2 et l'extrémité x_j de u_q non adjacent à u_{q-1} sont appelés les extrémités de la chaîne.

On dit que la chaîne L joint les sommets x_i et x_j .

Un arcs est une chaîne de longueur 1

Chaîne élémentaire :

C'est une chaîne tel quand la parcourant on ne rencontre pas deux fois le même sommet.

Cycle :

C'est une chaîne fermée.

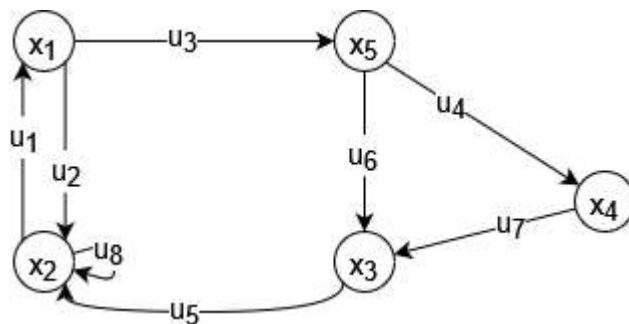
Cyclique élémentaire :

C'est un cycle minimal. C'est-à-dire ne contenant aucun autre cycle.

En parcourant un cycle élémentaire on ne rencontre pas deux fois le même sommet sauf le sommet de départ.

Une boucle est un cycle de longueur 1.

Exemple :



$$L = \{u_3; u_4\}$$

$$L = \{u_1; u_5; u_7\}$$

$$Cycle = \{u_6; u_7; u_4\}$$

1.4.2 Chemin et circuit

Chemin :

Un chemin $u = \langle u_1, u_2, \dots, u_q \rangle$ de cardinalité q **est une chaîne** de cardinalité q dont **les arcs sont orienté dans le même sens**. L'extrémité terminal de u_k coïncide avec l'extrémité initiale de u_{k+1} .

Si x_0 est l'extrémité initiale de u_1 et x_6 l'extrémité finale de u_q , on parle alors de chemin allant de x_a à x_b .

Circuit :

Un circuit est un chemin dont les extrémités coïncide. C'est aussi un cycle dont tous les arcs sont dirigé dans le même sens.

Le sous ensemble de sommet atteignable à partir d'un sommet donné grâce à des chemins est appelé fermeture transitive de ce sommet.

Parcours :

Dans les définitions suivantes, le terme parcours regroupe les chemins, les chaînes, les circuits et les cycles.

Un parcours est :

- **Élémentaire** lorsqu'en le parcourant on ne rencontre pas de fois le même sommets.
- **Simple** si tous les arcs qui le compose sont tous distinct.
- **Hamiltonien** lorsqu'il passe une et une seule fois par chaque sommet du graphe.
- **Eulérien** lorsqu'il passe une seule et une seule fois par chaque arc du graphe.
- **Pré-hamiltonien** lorsqu'il passe au moins une fois par chaque sommet du graphe.
- **Pré-eulérien** (ou chinois) lorsqu'il passe au moins une fois par arcs du graphe.

Théorème d'Euler :

Un graphe connexe G admet un cycle eulérien si et seulement si **tout ces sommets sont de degré pair**.

Théorème d'Euler 2 :

Un graphe connexe G admet une chaîne eulérienne distinct d'un cycle si et seulement si le **nombre de sommet de G de degré impair est égale à 2**.

Dans ce cas si A et B sont les deux sommet de G de degré impair alors le graphe G admet un chaîne eulérienne d'extrémité A et B .

1.4.3 Fermeture transitive d'un graphe

A tout graphe $G=(X, U)$ on peut associer de manière unique graphe transitif $\hat{G}=(X, \hat{U})$ appelé fermeture transitive de G où \hat{U} est définie par la relation d'appartenance suivante
 $(x, y) \in \hat{U} \Leftrightarrow$ il existe dans G un chemin [...]

Définition :

La fermeture ou clôture transitive d'un graphe G est le graphe aillant les même sommet que G et donc les arrêtes (arcs) relie x à y , s'il existe dans G une chaîne (chemin) de x à y .

Exemple

[G7]

On construit \hat{U} en ajoutant à U un arcs (x_i, x_k) qui n'appartient pas à U si $(x_i, x_j) \in U$ et $(x_j, x_k) \in U$ pour au moins un j et ce de façon itérative jusqu'à l'obtention d'un graphe \hat{G} transitif.

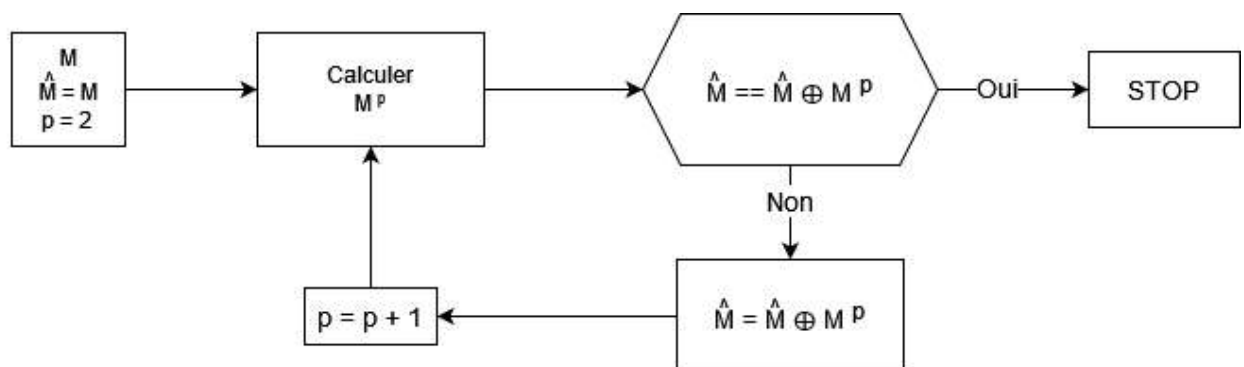
1.4.4 Algorithme d'obtention de la fermeture transitive d'un graphe

$M^p = M \otimes M \otimes M$ (p facteur)

\otimes produit booléen

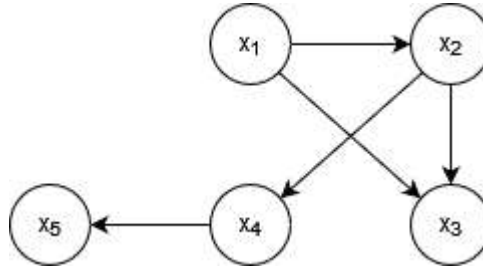
\oplus somme booléen

M = matrice d'adjacence



Exemple :

Pour plus de visibilité les "0" ont été remplacé par un "." .



$$M = \begin{pmatrix} . & 1 & 1 & . & . \\ . & . & 1 & 1 & . \\ . & . & . & . & . \\ . & . & . & . & 1 \\ . & . & . & . & . \end{pmatrix}$$

$$\hat{M} = M = \begin{pmatrix} . & 1 & 1 & . & . \\ . & . & 1 & 1 & . \\ . & . & . & . & . \\ . & . & . & . & 1 \\ . & . & . & . & . \end{pmatrix}$$

$$M^2 = \begin{pmatrix} . & . & 1 & 1 & . \\ . & . & . & . & 1 \\ . & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \end{pmatrix}$$

$$\hat{M} \oplus M^2 = \hat{M}_1 = \begin{pmatrix} . & 1 & 1 & 1 & . \\ . & . & 1 & 1 & 1 \\ . & . & . & . & . \\ . & . & . & . & 1 \\ . & . & . & . & . \end{pmatrix}$$

$$\hat{M} \neq \hat{M}_1 \Rightarrow p = p+1 = 3$$

Calculer M^3

$$M^3 = M \otimes M^2 = \begin{pmatrix} . & . & . & . & 1 \\ . & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \end{pmatrix}$$

$$\hat{M} \oplus M^3 = \hat{M}_2 = \begin{pmatrix} . & 1 & 1 & 1 & 1 \\ . & . & 1 & 1 & 1 \\ . & . & . & . & . \\ . & . & . & . & 1 \\ . & . & . & . & 1 \end{pmatrix}$$

$$\hat{M}_2 = \begin{pmatrix} . & 1 & 1 & 1 & 1 \\ . & . & 1 & 1 & 1 \\ . & . & . & . & . \\ . & . & . & . & 1 \\ . & . & . & . & . \end{pmatrix}$$

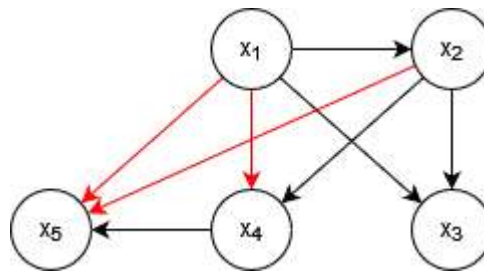
$$\hat{M}_2 \neq \hat{M}_1 \Rightarrow p = p+1 = 4$$

Calculer M^4

$$M^4 = M \otimes M^3 = \begin{pmatrix} . & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \end{pmatrix} \text{ Matrice nulle}$$

$$\hat{M}_3 = \hat{M}_2 \oplus M^4 = \hat{M}_2$$

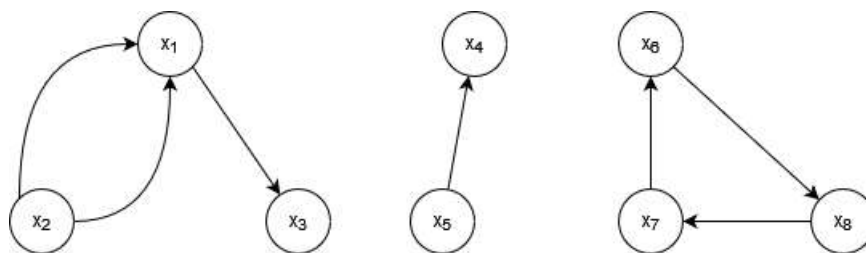
$$\hat{M}_3 = \hat{M}_2 \Rightarrow \text{STOP}$$



1.4.5 Connexité, nombre de connexité

Un graphe $G=(X, U)$ est connexe si quelque soit $\forall x_i \in X, \forall x_j \in X, x_i \neq x_j$ il existe une chaîne entre (x_i, x_j) on appelle composante connexe le sous ensemble de sommet tel qu'il existe une chaîne entre deux sommet quelconque distinct. Un graphe est connexe s'il comporte une composante et une seul. Chaque composante connexe est un graphe connexe.

Exemple :



Graphes ayant 3 composante connexes

1.4.6 Forte connexité

Un graphe $G=(X, U)$ est fortement connexe si $\forall x_i \in X, \forall x_j \in X, x_i \neq x_j$, il existe au moins un chemin de x_i à x_j et un chemin joignant x_j à x_i .

Une composante fortement connexe (CFC) est un sous ensemble de sommet tel qu'il existe un chemin entre deux sommets quelconque. Une composante fortement connexe maximal (CFCM) est un ensemble maximal de CFC. Les différentes CFCM définissent une partition de X . Un graphe est fortement connexe s'il ne comprend qu'une seule composante fortement connexe maximal.

Algorithme de recherche CFCM :

Principe :

- Déterminer une composante f-connexe C, en partant d'un sommet quelconque.
- Retirer C du graphe et recommencer

X = Ensemble de sommet

$E = X$

Tant que(1) E non Vide Faire :

marquer + et - un sommet (quelconque) de E ;

Tant que(2) c'est possible Faire :

marquer + tout **successeur** d'un sommet déjà marqué + ;

marquer - tout **prédécesseur** d'un sommet déjà marqué - ;

Fin tant que(2)

Écrire C l'ensemble des sommets marqués + et - ;

Le sous-graphe de G dont les sommets sont ceux de C est une composante f-connexe de G ;

$E = E - C$

$C = \emptyset$

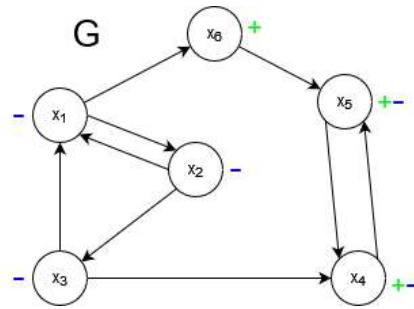
Fin Tant que(1)

FIN

Exemple :

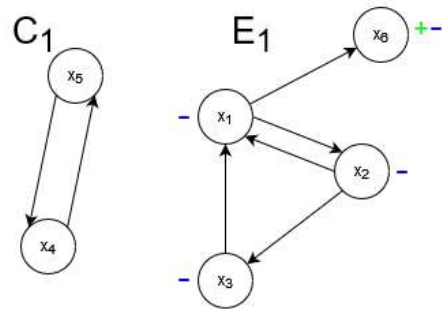
$$X = \{x_1, x_2, x_3, x_4, x_5, x_6\}$$

$$E = X = \{x_1, x_2, x_3, x_4, x_5, x_6\} \quad \text{with a red dot on } x_6$$



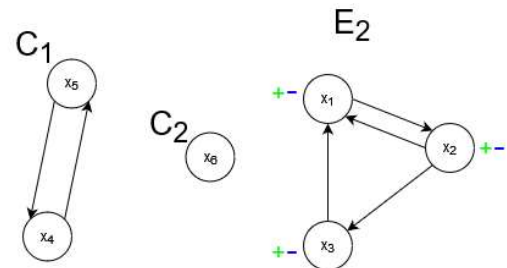
$$C_1 = \{x_4, x_5\}$$

$$E_1 = E - C_1 = \{x_1, x_2, x_3, x_6\}$$



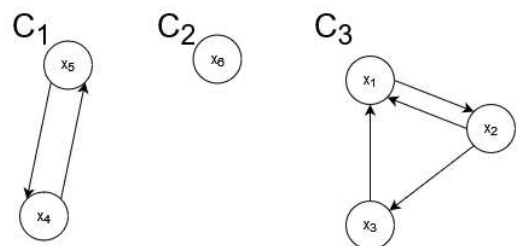
$$C_2 = \{x_6\}$$

$$E_2 = E_1 - C_2 = \{x_1, x_2, x_3\}$$

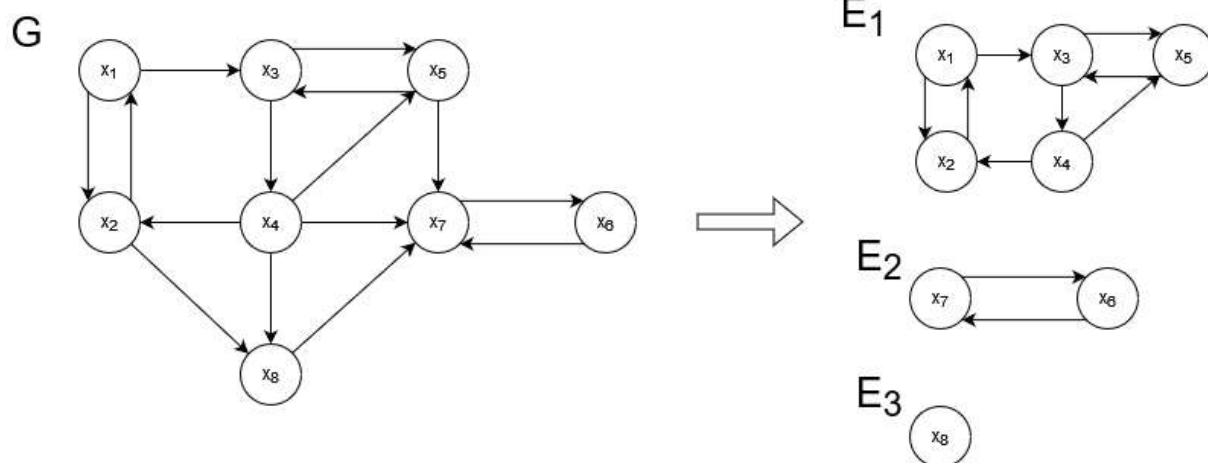


$$C_3 = \{x_1, x_2, x_3\}$$

$$E_3 = E_2 - C_3 = \emptyset$$



Exemple 2 :



1.5 Graphes particulier

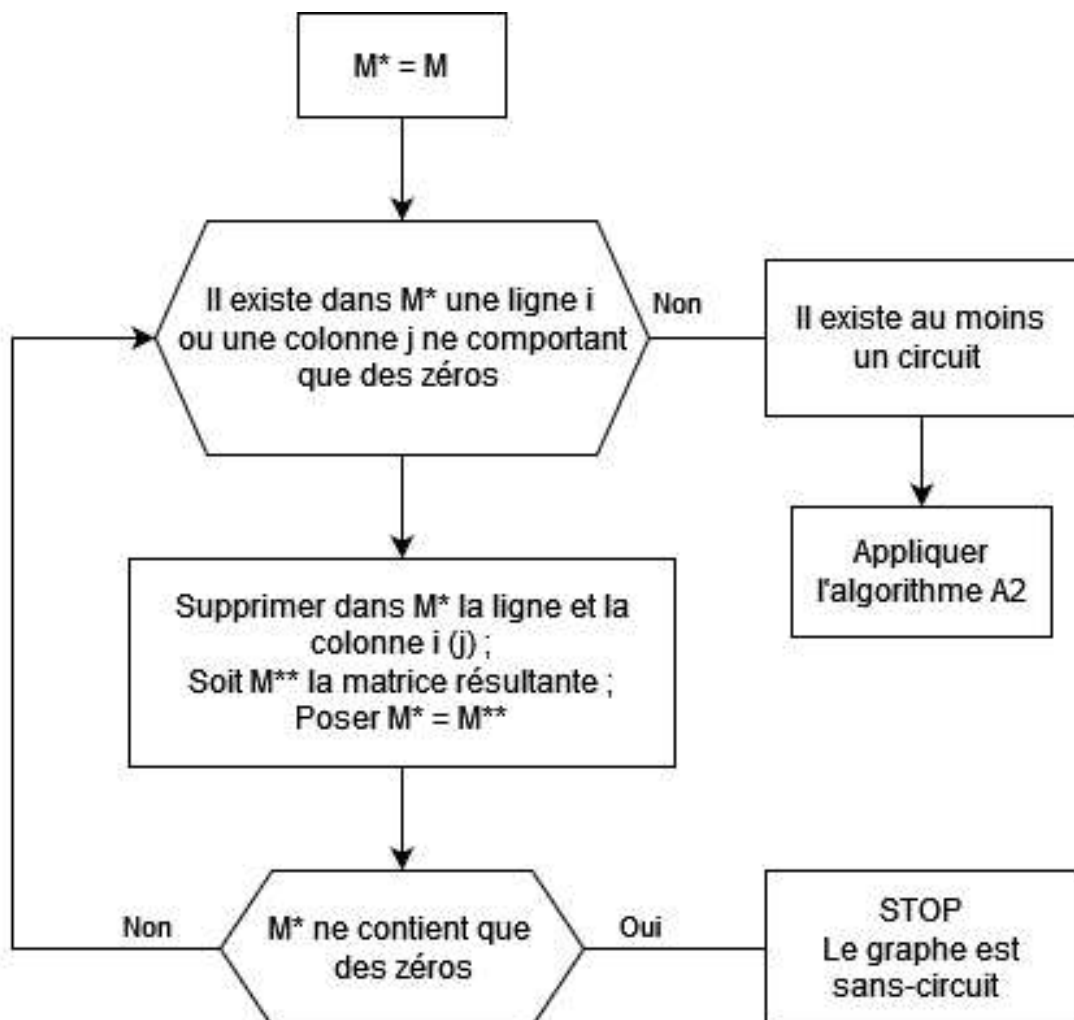
1.5.1 Graphes sans circuits

L'absence de circuit est indispensable à l'utilisation d'un grand nombre d'algorithmes. On les rencontre lors de la représentation d'une relation d'ordre sur des éléments.

A partir de certain sommet, on ne peut atteindre qu'un sous-ensemble de sommets ou alors aucun des sommets de ce sous-ensemble ne peut atteindre les sommets de départ. Les différents sous-ensembles constituent des niveaux ordonnés par un **rang** (le rang est la distance maximale entre d'un sommet à la racine).

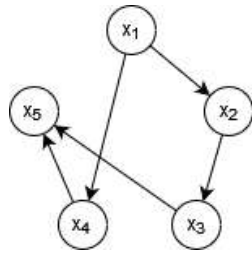
Il n'existe pas de chemin allant d'un sommet à un autre situé à un même niveau ou dans un niveau de rang inférieur.

A1 :



Exemple :

1 :



$$M_a = \begin{pmatrix} . & 1 & . & 1 & . \\ . & . & 1 & 1 & . \\ . & . & . & . & 1 \\ . & . & . & . & 1 \\ . & . & . & . & . \end{pmatrix}$$

$$M_a^* = \begin{pmatrix} . & 1 & 1 & . \\ . & . & 1 & . \\ . & . & . & 1 \\ . & . & . & 1 \\ . & . & . & . \end{pmatrix} \quad (1^\circ)$$

$$M_a^* = \begin{pmatrix} . & 1 & 1 & . \\ . & . & . & 1 \\ . & . & . & 1 \\ . & . & . & . \end{pmatrix}$$

$$M_a^* = \begin{pmatrix} . & 1 & 1 & . \\ . & . & . & 1 \\ . & . & . & 1 \\ . & . & . & . \end{pmatrix} \quad (2^\circ)$$

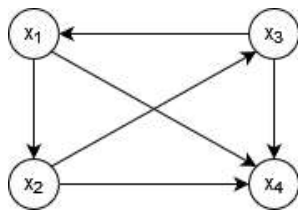
$$M_a^{**} = \begin{pmatrix} . & 1 & 1 \\ . & . & . \\ . & . & . \end{pmatrix}$$

$$M_a^* = \begin{pmatrix} . & 1 & 1 \\ . & . & . \\ . & . & . \end{pmatrix} \quad (3^\circ)$$

$$M_a^{**} = \begin{pmatrix} . & . \\ . & . \end{pmatrix} = \text{Matrice nul} \Rightarrow \text{pas de circuit}$$

Les numéros en rouge indiquent l'ordre de suppression des lignes/colonnes.
Il est important de les mettre pour mieux se retrouver.

2 :



$$M_b = \begin{pmatrix} . & 1 & . & 1 \\ . & . & 1 & 1 \\ 1 & . & . & 1 \\ . & . & . & . \end{pmatrix}$$

$$M_b^* = \begin{pmatrix} . & 1 & . & 1 \\ . & . & 1 & 1 \\ 1 & . & . & 1 \\ . & . & . & . \end{pmatrix} \quad (1^\circ)$$

$$M_b^{**} = \begin{pmatrix} . & 1 & . \\ . & . & 1 \\ 1 & . & . \end{pmatrix}$$

Il n'existe plus de ligne ne comportant que des zéros. Cela signifie qu'il existe un circuit.

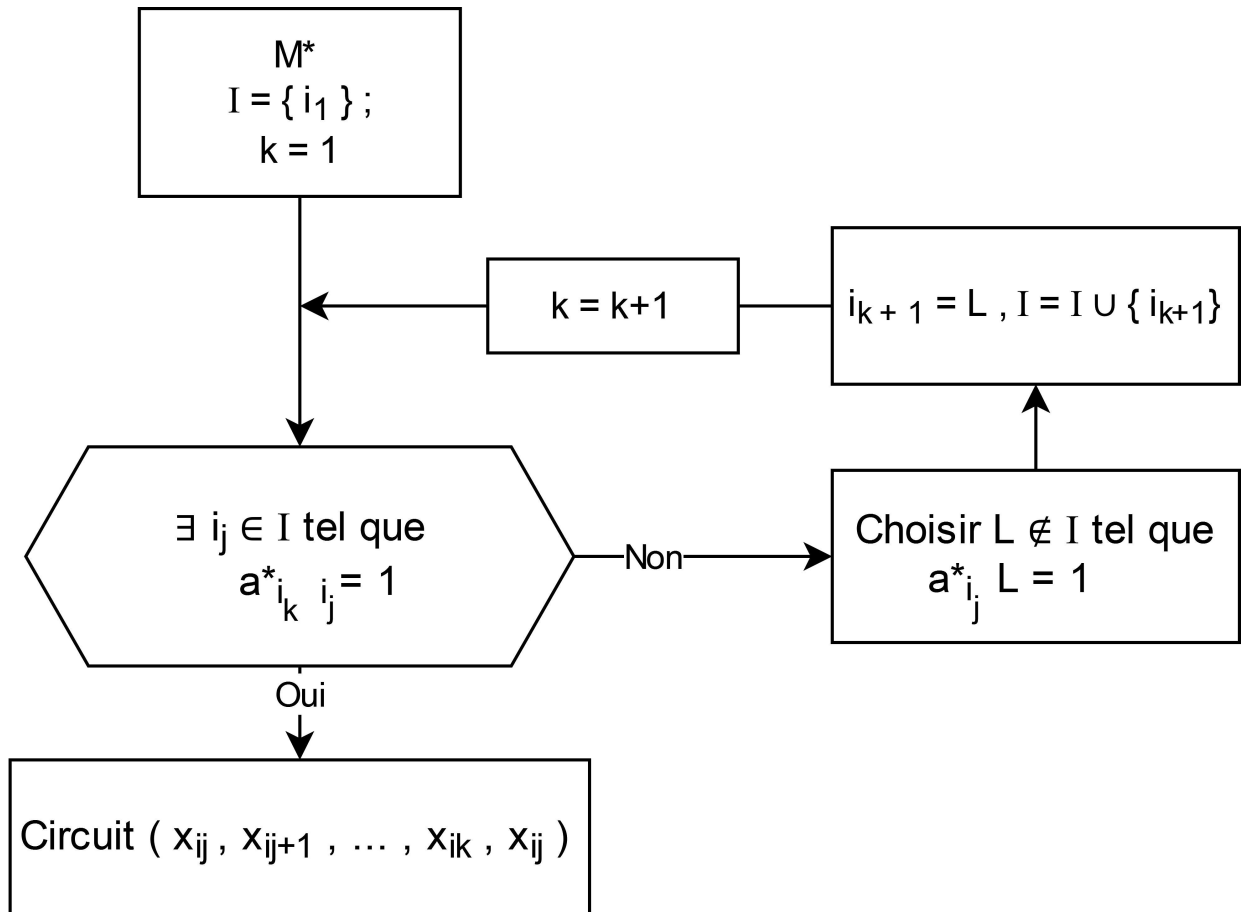
1.5.2 Algorithme d'obtention d'un circuit dans un graphes (A2)

Lorsque le graphe possède au moins un circuit, on peut en construire un au moyen de l'algorithme A2.

On part d'un sommet quelconque du sous graphe G^* défini par la matrice $M^* = (a_{ij}^*)$ obtenue lors de la dernière itération de l'algorithme A1.

[A2]

A2 :



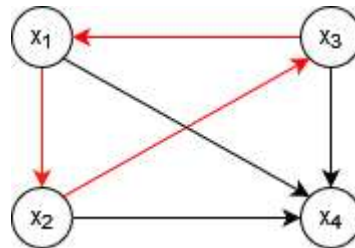
- i_1 représente un sommet quelconque graphe.
- i_j est un sommet.
- $a_{i_k i_j}^* = 1$ est un arc avec i_k comme sommet initial et i_j comme sommet final. Le 1 fait référence à la matrice d'adjacence où pour que l'arc existe un 1 doit se trouver à la ligne k colonne j .
- Dans cette représentation de l'algorithme nous avons remplacé le "l" minuscule en "L" majuscule pour éviter les confusions avec le chiffre "1".

Exemple :

Nous allons rechercher le circuit du graphe de l'exemple précédent.

Initialisation : $M^* = \begin{pmatrix} \cdot & 1 & \cdot \\ \cdot & \cdot & 1 \\ 1 & \cdot & \cdot \end{pmatrix}; I = \{x_2\}; k=1$

Choisir $L = x_3$; $i_{k+1} = L = x_3$; $I = I \cup \{i_{k+1}\} = I \cup \{x_3\} = \{x_2, x_3\}$	$k = k + 1 = 2$
Choisir $L = x_2$; $i_{k+1} = L = x_2$; $I = I \cup \{i_{k+1}\} = I \cup \{x_1\} = \{x_2, x_3, x_1\}$	$k = k + 1 = 3$
$a_{x_1 x_2}^* = 1 \Rightarrow \text{STOP}$	Circuit(x_2, x_3, x_1, x_2)



2-Problème de cheminement dans les graphes

en particulier la recherche d'un plus court chemin compte parmi les problèmes les plus anciens de la théorie des graphes et les plus importants par leur application.

2.1-Définition

En tant donné un graphe $G=(X, U)$ on associe à chaque arc $u=(x_i, x_j)$ élément de U un nombre $l(u) \in \mathbb{R}$ appelé longueur de l'arc ou l_{ij} . On dit que G est valué par les longueurs $l(u)$.

Le problème du plus court chemin entre deux sommets x_i et x_j est de trouver un chemin $\mu(x_i, x_j)$ tel que $l(\mu) = \sum_{u \in \mu} l(u)$ soit minimale. $l(\mu)$ peut être un coût de transport, une dépense de construction, un temps nécessaire de parcours, ...

2.2 Recherche de chemins minimaux dans les graphes

Principes d'optimalité

le principe d'optimalité énonce que les sous chemins des plus courts chemins sont des plus courts chemins.

Lemme :

Soit un graphe $G=(X, U)$ et la fonction $P: X \times X \rightarrow \mathbb{R}$ un plus court chemin de x_1 à x_k

Soit $C = \langle x_1, x_2, \dots, x_k \rangle$

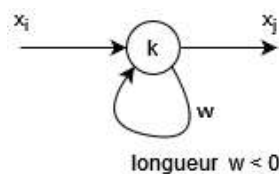
$\forall (i, j) \text{ tel que } 1 \leq i \leq j \leq k$

Soit $C_{ij} = \langle x_i, x_{i+1}, \dots, x_j \rangle$ Un sous-chemin de x_i à x_j de C

alors C_{ij} est un plus court chemin de x_i à x_j .

Remarque si dans le graphe il existe un circuit de longueur négative, la recherche d'un plus court chemin de l'origine au sommet considéré est sans objet (impossible).

On peut utiliser le circuit une infinité de fois.



2.3-Algorithmme de résolution

Algorithme de FORD :

Peu efficace, il permet l'obtention de chemin de longueur minimal et maximal quelque soit les signes des longueur l_{ij} .

Principe :

Plus court chemin :

On marque tout sommet x_i d'une marque λ_i qui correspond à une borne supérieure pour la distance minimale de x_i à x_j . On diminue progressivement ces marque à raison d'une par étape et en suivant la procédure écrite dans l'organigramme.

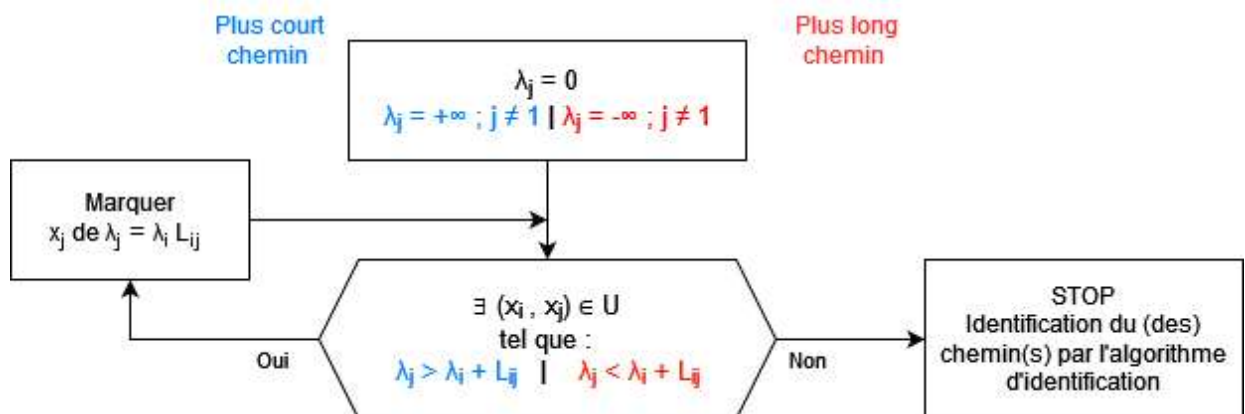
Quand une tel diminution n'est plus possible la marque λ_j est la distance minimal de x_i à x_j .

Plus long chemin :

On marque tout sommet x_i d'une marque λ_i qui correspond à une borne inférieure pour la distance minimale de x_i à x_j . On augmente progressivement ces marques à raison d'une par étape et en suivant la procédure écrite dans l'organigramme.

Quand une tel augmentation n'est plus possible la marque λ_j est la distance maximal de x_i à x_j .

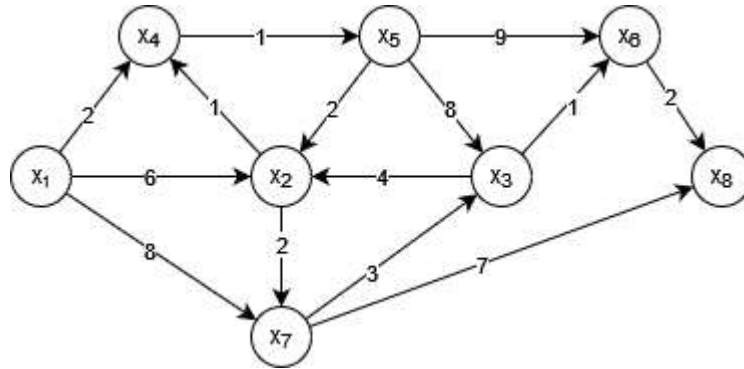
Un chemin de longueur minimale (maximale) peut donc être obtenue à partir des distance λ_j ainsi obtenue par application des algorithmes d'identification.



Inconvénient :

Si le sommet marqué à chaque étape est choisie au hasard, le nombre d'itérations peut être élevé.

Exemple :



$$L = \begin{pmatrix} 0 & 6 & . & 2 & . & . & 8 & . \\ . & 0 & . & 1 & . & . & 2 & . \\ . & 4 & 0 & . & . & 1 & . & . \\ . & . & . & 0 & 1 & . & . & . \\ . & 2 & 8 & . & 0 & 9 & . & . \\ . & . & . & . & . & 0 & . & 2 \\ . & . & 3 & . & . & . & 0 & 7 \\ . & . & . & . & . & . & . & 0 \end{pmatrix}$$

Pour plus de clarté " ∞ " à été remplacé par un ".".

En partant de x_1 voici les distances jusqu'aux différent sommet du graphes :

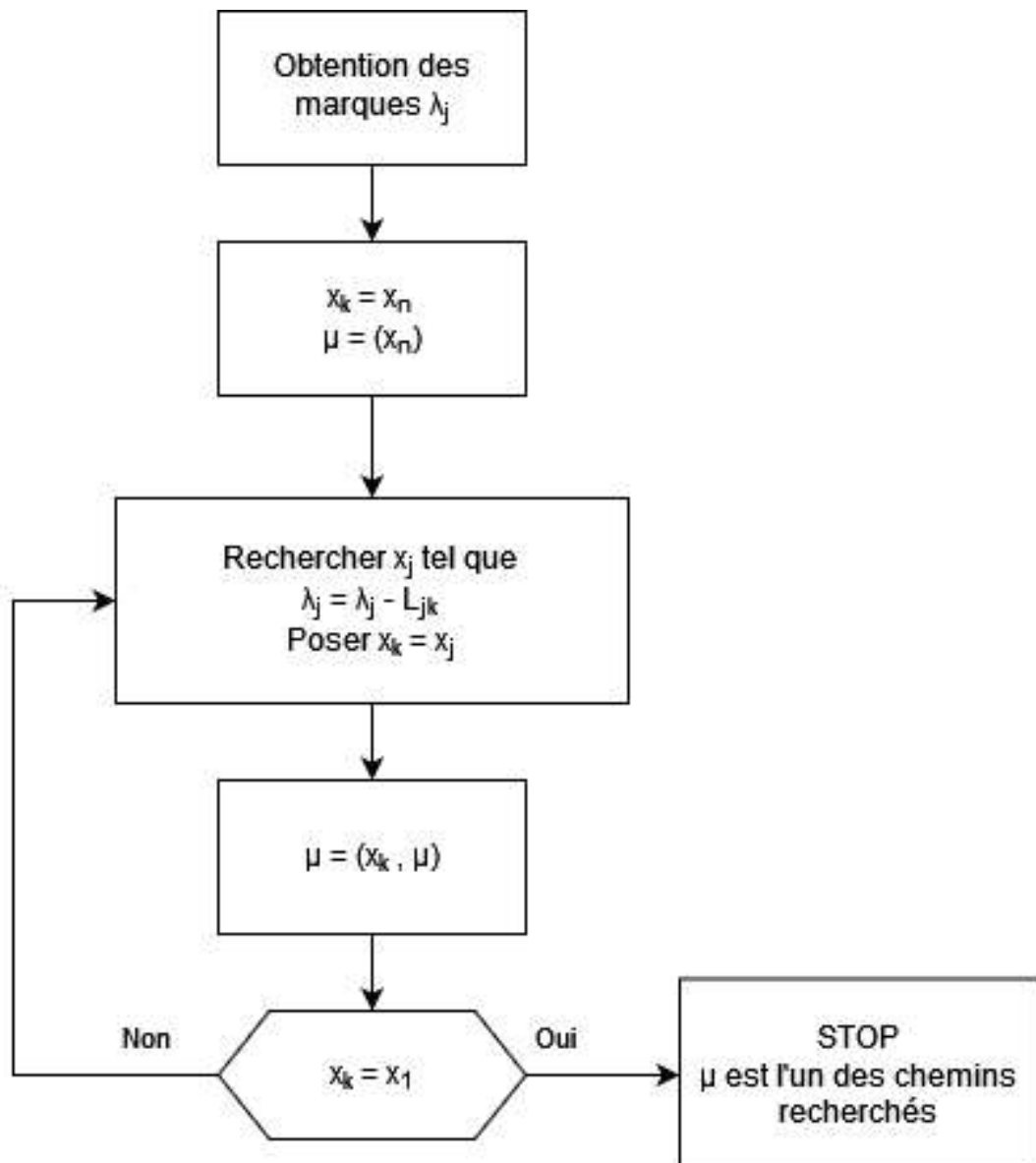
Sommet		x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8		
Marques initiales		0	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$		
Arcs pris en compte pour modifier les marques	(x_1, x_4)	0	$+\infty$	$+\infty$	0+2=2	$+\infty$	$+\infty$	$+\infty$	$+\infty$		
	(x_1, x_2)		0+6=6		2						
	(x_1, x_7)		6					$2+1=3$		0+8=8	
	(x_4, x_5)		$3+2=5$		3			8			
	(x_5, x_2)							$5+2=7$			
	(x_2, x_7)		5					$7+3=10$			
	(x_7, x_3)				$10+1=11$	7					
	(x_3, x_6)			10	11	$11+2=13$					
	(x_6, x_8)										
Marques finales		0	5	10	2	3	11	7	13		

2.3.2 – Algorithme d'identification d'un chemin de longueur minimal (maximal)

L'identification d'un chemin de longueur minimal (maximal) se fait à partir des distances minimale (maximale) de x_1 à x_j . Les distances correspondent aux marques λ_j obtenues en fin d'application des algorithmes de FORD ou de Dijkstra.

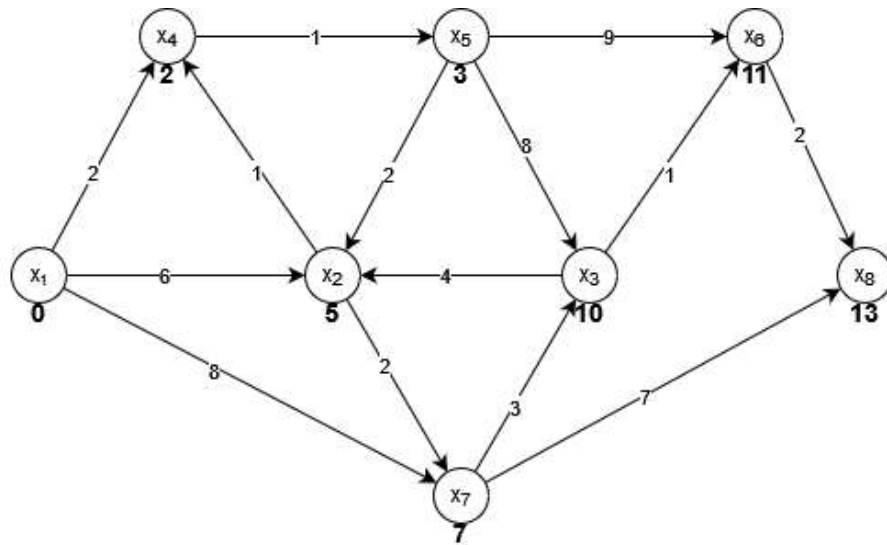
Principe :

La construction des chemins recherchés se fait à reculons à partir de x_n (i.e. l'arrivé) par juxtaposition d'arcs (x_j, x_k) tels que $\lambda_j = \lambda_k - l_{jk}$.



Exemple :

Application de l'algorithme sur le graphe précédent.



On part de x_8 $\mu = (x_8)$

$13 - 2 = 11$ *c est la marque de $x_6 \Rightarrow$ on prend x_6*

$13 - 7 = 6$

$\mu = (x_6, \mu) = (x_6, x_8)$

On part de x_6

$11 - 9 = 2$

$11 - 1 = 10$ *c est la marque de $x_3 \Rightarrow$ on prend x_3*

$\mu = (x_3, \mu) = (x_3, x_6, x_8)$

On part de x_3

$10 - 8 = 2$

$10 - 3 = 7$ *c est la marque de $x_7 \Rightarrow$ on prend x_7*

$\mu = (x_7, \mu) = (x_7, x_3, x_6, x_8)$

On part de x_7

$7 - 2 = 5$ *c est la marque de $x_2 \Rightarrow$ on prend x_2*

$7 - 8 = -1$

$\mu = (x_2, \mu) = (x_2, x_7, x_3, x_6, x_8)$

On part de x_2

$5-2=3$ *c 'est la marque de $x_5 \Rightarrow$ on prend x_5*

$5-6=-1$

$5-4=1$

$\mu=(x_5, \mu)=(x_5, x_2, x_7, x_3, x_6, x_8)$

On part de x_5

$3-1=2$ *c 'est la marque de $x_4 \Rightarrow$ on prend x_4*

$\mu=(x_4, \mu)=(x_4, x_5, x_2, x_7, x_3, x_6, x_8)$

On part de x_4

$2-1=1$

$2-2=0$ *c 'est la marque de $x_1 \Rightarrow$ on prend x_1*

$\mu=(x_1, \mu)=(x_1, x_4, x_5, x_2, x_7, x_3, x_6, x_8)$

$x_1=x_1 \Rightarrow STOP$

$\mu=(x_1, x_4, x_5, x_2, x_7, x_3, x_6, x_8)$

Algorithme de Dijkstra-Moore

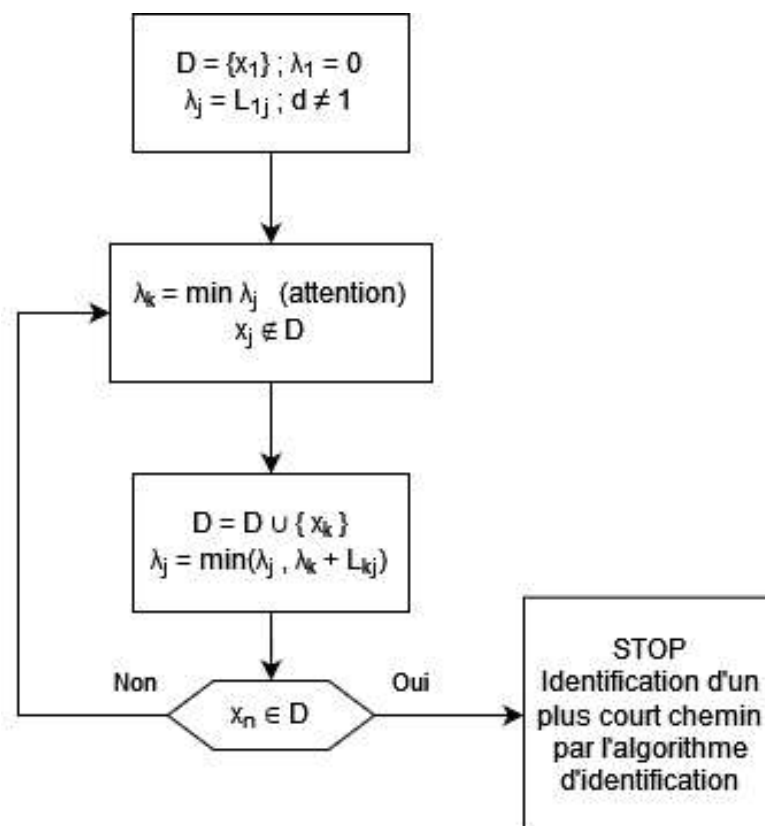
Cet algorithme permet uniquement l'obtention de chemin de longueur minimal et sous la condition que toute les longueur L_{ij} soit positive.

Principe :

On considère au départ de x_1 les chemins d'un arcs (étape 1) puis ceux de deux arcs au maximum (étape 2), ... et ceux de t arcs (étape t).

Soit D l'ensemble des sommets marqués jusqu'à l'étape $t - 1$. A l'étape t on considère l'ensemble des sommets n'appartenant pas à D mais ayant un successeur dans D et on marque l'un sommet, soit x_k d'une marque définitive λ_k qui représente la distance minimal de x_1 à x_k .

Lorsque le sommet x_n est marqué les λ_j obtenue représente les distances de x_1 à x_j nécessaire à l'algorithme d'identification.



(attention) :

S'il arriverait que plusieurs $\lambda_k = \lambda_{k_1}, \lambda_{k_2}, \dots, \lambda_{k_l}$ réalisent ce minimum,

tous sont à ajouter à D l'instruction devient $\lambda_j = \min(\lambda_j, \lambda_{k_1} + L_{k_1j}, \dots, \lambda_{k_l} + L_{k_lj})$

Exemple :

D	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
$D=\{x_1\}$	0	6	$+\infty$	2	$+\infty$	$+\infty$	8	$+\infty$
$D=\{x_1, x_4\}$	0	6	$+\infty$	2	$2+1=3$	$+\infty$	8	$+\infty$
$D=\{x_1, x_4, x_5\}$	0	$2+3=5$	$3+8=11$	2	3	$3+9=12$	8	$+\infty$
$D=\{x_1, x_4, x_5, x_2\}$	0	5	11	2	3	12	$5+2=7$	$+\infty$
$D=\{x_1, x_4, x_5, x_2, x_7\}$	0	5	$7+3=10$	2	3	12	7	$7+7=14$
$D=\{x_1, x_4, x_5, x_2, x_7, x_3\}$	0	5	10	2	3	$10+1=11$	7	14
$D=\{x_1, x_4, x_5, x_2, x_7, x_3, x_6\}$	0	5	10	2	3	11	7	$11+2=13$
$D=\{x_1, x_4, x_5, x_2, x_7, x_3, x_6, x_8\}$	0	5	10	2	3	11	7	13
Marques finales	0	5	10	2	3	11	7	13