

# Introduction à la théorie des graphes



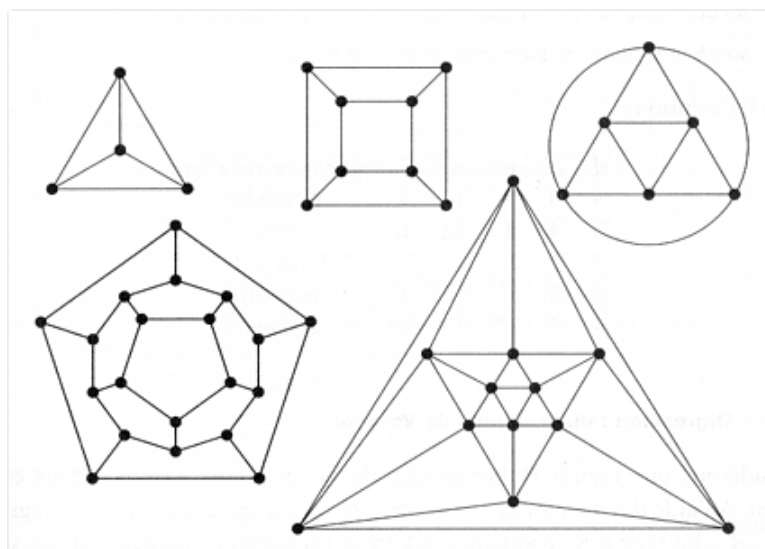
Didier Müller, 2008

[www.apprendre-en-ligne.net/graphes](http://www.apprendre-en-ligne.net/graphes)



# Table des matières

1. Graphes non orientés.....	2
2. Graphe partiel et sous-graphe.....	4
3. Degrés.....	5
4. Chaînes et cycles.....	6
5. Graphes eulériens.....	7
6. Graphes hamiltoniens.....	9
7. Graphes planaires.....	9
8. Matrice et listes d'adjacences.....	10
9. Arbres.....	11
10. Codage de Prüfer.....	12
11. Arbres couvrants.....	15
12. Arborescences.....	16
13. Codage de Huffman.....	17
14. Coloration des sommets.....	19
15. Coloration des sommets d'un graphe planaire.....	22
16. Coloration des arêtes.....	23
17. Graphes orientés (digraphes).....	23
18. Degré d'un sommet d'un digraphe.....	24
19. Chemins et circuits.....	24
20. Matrice et listes d'adjacences.....	25
21. Digraphes sans circuits.....	26
22. Algorithme de Dijkstra.....	27
23. Méthode PERT.....	29
24. Introduction aux chaînes de Markov.....	31
25. Distribution limite.....	32
26. Chaîne absorbante.....	34
Lexique sur les graphes.....	37
Quelques références sur la théorie des graphes.....	44



# 1. Graphes non orientés

Un **graphe** fini  $G = (V, E)$  est défini par l'ensemble fini  $V = \{v_1, v_2, \dots, v_n\}$  ( $|V| = n$ ) dont les éléments sont appelés **sommets**, et par l'ensemble fini  $E = \{e_1, e_2, \dots, e_m\}$  ( $|E| = m$ ) dont les éléments sont appelés **arêtes**.

Une arête  $e$  de l'ensemble  $E$  est définie par une paire **non-ordonnée** de sommets, appelés les extrémités de  $e$ . Si l'arête  $e$  relie les sommets  $a$  et  $b$ , on dira que ces sommets sont **adjacents**, ou **incidents** avec  $e$ , ou encore que l'arête  $e$  est **incidente** avec les sommets  $a$  et  $b$ .

On appelle **ordre d'un graphe** le nombre de sommets ( $n$ ) de ce graphe.

## Représentation graphique

Les graphes tiennent leur nom du fait qu'on peut les représenter par des dessins. À chaque sommet de  $G$ , on fait correspondre un point distinct du plan et on relie par une courbe simple les points correspondant aux extrémités de chaque arête.

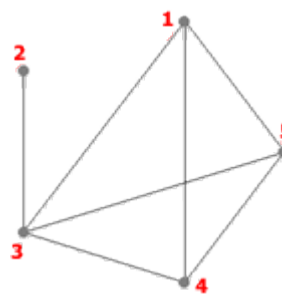
### Exemple

Il existe une infinité de manières de représenter graphiquement un graphe. Vous voyez ci-contre deux représentations graphiques du même graphe  $G = (V, E)$  décrit ci-dessous par l'ensemble de ses sommets et l'ensemble de ses arêtes.

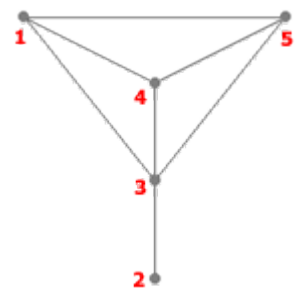
Ensemble des sommets :  $V = \{1, 2, 3, 4, 5\}$

Ensemble des arêtes :

$E = \{(1, 3), (1, 4), (1, 5), (2, 3), (3, 4), (3, 5), (4, 5)\}$



Une représentation non planaire de  $G$



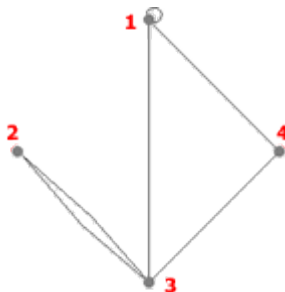
Une représentation planaire de  $G$

## Quelques types de graphes

Si on arrive à dessiner le graphe sans qu'aucune arête n'en coupe une autre (les arêtes ne sont pas forcément rectilignes), on dit que le graphe est **planaire**.

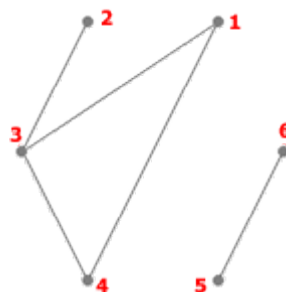
Les graphes ci-dessus sont **simples**, mais on peut imaginer des graphes avec une arête qui relie un sommet à lui-même (une boucle), ou plusieurs arêtes reliant les deux mêmes sommets (voir ci-dessous, à gauche). Dans ce cas, on parle de **multigraphe**.

Un graphe est **connexe** s'il est possible, à partir de n'importe quel sommet, de rejoindre tous les autres en suivant les arêtes. Un graphe non connexe se décompose en **composantes connexes**. Sur le graphe ci-dessous, au milieu, les composantes connexes sont  $\{1, 2, 3, 4\}$  et  $\{5, 6\}$ .



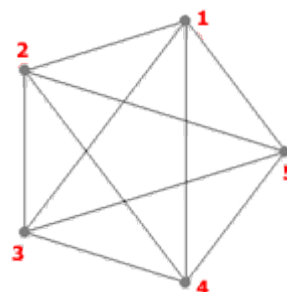
Graphe non simple (multigraphe)

$V = \{1, 2, 3, 4\}$   
 $E = \{(1,1), (1,3), (1,4), (2,3), (2,3), (3,4)\}$



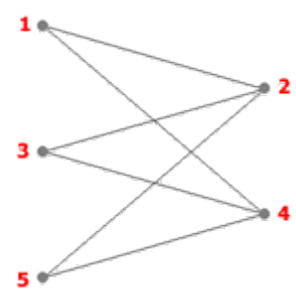
Graphe non connexe

$V = \{1, 2, 3, 4, 5, 6\}$   
 $E = \{(1,3), (1,4), (2,3), (3,4), (5,6)\}$



Graphe complet  $K_5$

$V = \{1, 2, 3, 4, 5\}$   
 $E = \{(1,2), (1,3), (1,4), (1,5), (2,3), (2,4), (2,5), (3,4), (3,5), (4,5)\}$



Graphe biparti

$V = \{1, 2, 3, 4, 5\}$   
 $E = \{(1,2), (1,4), (2,3), (2,5), (3,4), (4,5)\}$

Un graphe est **complet** si chaque sommet du graphe est relié directement à tous les autres sommets (voir ci-dessus).

Un graphe est **biparti** si ses sommets peuvent être divisés en deux ensembles  $X$  et  $Y$ , de sorte que toutes les arêtes du graphe relient un sommet dans  $X$  à un sommet dans  $Y$  (dans l'exemple ci-dessus à droite, on a  $X = \{1, 3, 5\}$  et  $Y = \{2, 4\}$ , ou vice versa).

### Exercice 1.1

On a 6 wagons à trier. Dans la gare de triage, les wagons entrent dans l'ordre 2, 5, 3, 6, 1, 4 et doivent sortir dans l'ordre croissant. Deux wagons  $i$  et  $j$  peuvent être mis sur la même voie si et seulement s'ils entrent dans l'ordre dans lequel ils doivent sortir.

Dessinez un graphe illustrant la situation, en indiquant ce que représentent les sommets et les arêtes de votre graphe.

Quel sera le nombre minimal de voies nécessaires au tri ?

### Exercice 1.2

Trois professeurs P1, P2, P3 doivent donner ce lundi un certain nombre d'heures de cours à trois classes C1, C2, C3:

- P1 doit donner deux heures de cours à C1 et une heure à C2 ;
- P2 doit donner une heure de cours à C1, une heure à C2 et une heure à C3 ;
- P3 doit donner une heure de cours à C1, une heure à C2 et deux heures à C3.

Comment représenter cette situation par un graphe ?

Quel type de graphe obtenez-vous ?

Combien faudra-t-il de plages horaires au minimum ?

Aidez-vous du graphe pour proposer un horaire du lundi pour ces professeurs.

### Exercice 1.3

Un tournoi d'échecs oppose 6 personnes. Chaque joueur doit affronter tous les autres.

Construisez un graphe représentant toutes les parties possibles.

Quel type de graphe obtenez-vous ?

Si chaque joueur ne joue qu'un match par jour, combien de jours faudra-t-il pour terminer le tournoi ?

Aidez-vous du graphe pour proposer un calendrier des matches.

### Exercice 1.4



Sur un échiquier 3x3, les deux cavaliers noirs sont placés sur les cases a1 et c1, les deux cavaliers blancs occupant les cases a3 et c3.

Aidez-vous d'un graphe pour déterminer les mouvements qui permettront aux cavaliers blancs de prendre les places des cavaliers noirs, et vice versa.

### Exercice 1.5

Un jour, Sherlock Holmes reçoit la visite de son ami Watson que l'on avait chargé d'enquêter sur un assassinat mystérieux datant de plus de trois ans.

À l'époque, le Duc de Densmore avait été tué par l'explosion d'une bombe, qui avait entièrement détruit le château de Densmore où il s'était retiré. Les journaux d'alors relataient que le testament, détruit lui aussi dans l'explosion, avait tout pour déplaire à l'une de ses sept ex-épouses. Or, avant de mourir, le Duc les avait toutes invitées à passer quelques jours dans sa retraite écossaise.

– *Holmes* : Je me souviens de cette affaire ; ce qui est étrange, c'est que la bombe avait été fabriquée spécialement

pour être cachée dans l'armure de la chambre à coucher, ce qui suppose que l'assassin a nécessairement effectué plusieurs visites au château !

- *Watson* : Certes, et pour cette raison, j'ai interrogé chacune des femmes : Ann, Betty, Charlotte, Edith, Félicia, Georgia et Helen. Elles ont toutes juré qu'elles n'avaient été au château de Densmore qu'une seule fois dans leur vie.
- *Holmes* : Hum ! Leur avez-vous demandé à quelle période elles ont eu leur séjour respectif ?
- *Watson* : Hélas ! Aucune ne se rappelait les dates exactes, après plus de trois ans ! Néanmoins, je leur ai demandé qui elles avaient rencontré :
  - Ann a rencontré Betty, Charlotte, Félicia et Georgia.
  - Betty a rencontré Ann, Charlotte, Edith, Félicia et Helen.
  - Charlotte a rencontré Ann, Betty et Edith.
  - Edith a rencontré Betty, Charlotte et Félicia.
  - Félicia a rencontré Ann, Betty, Edith et Helen.
  - Georgia a rencontré Ann et Helen.
  - Helen a rencontré Betty, Félicia et Georgia.

Vous voyez, mon cher Holmes, les réponses sont concordantes !

C'est alors que Holmes prit un crayon et dessina un étrange petit dessin, avec des points marqué A, B, C, E, F, G, H et des lignes reliant certains de ces points. Puis, en moins de trente secondes, Holmes déclara :

- Tiens, tiens ! Ce que vous venez de me dire détermine d'une façon unique l'assassin.

Qui est l'assassin ?

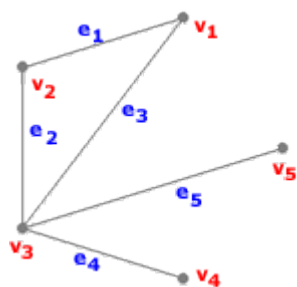
## 2. Graphe partiel et sous-graphe

Soit  $G = (V, E)$  un graphe. Le graphe  $G' = (V, E')$  est un **graphe partiel** de  $G$ , si  $E'$  est inclus dans  $E$ . Autrement dit, on obtient  $G'$  en enlevant une ou plusieurs arêtes au graphe  $G$ .

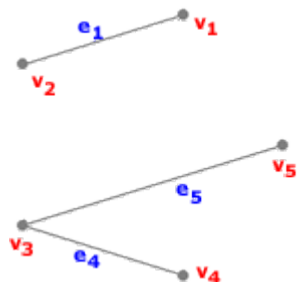
Pour un sous-ensemble de sommets  $A$  inclus dans  $V$ , le **sous-graphe** de  $G$  induit par  $A$  est le graphe  $G = (A, E(A))$  dont l'ensemble des sommets est  $A$  et l'ensemble des arêtes  $E(A)$  est formé de toutes les arêtes de  $G$  ayant leurs deux extrémités dans  $A$ . Autrement dit, on obtient  $G'$  en enlevant un ou plusieurs sommets au graphe  $G$ , ainsi que toutes les arêtes incidentes à ces sommets.

Un graphe partiel d'un sous-graphe est un **sous-graphe partiel** de  $G$ . On appelle **clique** un sous-graphe complet de  $G$ .

On appelle **stable** un sous-graphe de  $G$  sans arêtes.

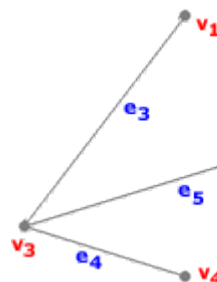


Soit le graphe  $G = (V, E)$   
 $V = \{v_1, v_2, v_3, v_4, v_5\}$   
 $E = \{e_1=(v_1, v_2), e_2=(v_2, v_3), e_3=(v_1, v_3), e_4=(v_3, v_4), e_5=(v_3, v_5)\}$



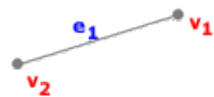
Graphe partiel de  $G$

$V' = V$   
 $E' = \{e_1, e_4, e_5\}$



Sous-graphe de  $G$

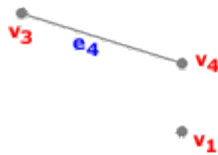
$V' = \{v_1, v_3, v_4, v_5\}$   
 $E' = \{e_3, e_4, e_5\}$



Sous-graphe partiel de  $G$

$$V' = \{v_1, v_2, v_3, v_4\}$$

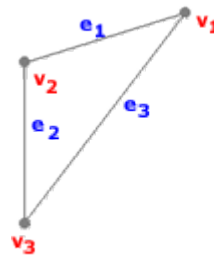
$$E' = \{e_1, e_4\}$$



Un stable de  $G$

$$V' = \{v_1, v_4, v_5\}$$

$$E' = \{\}$$



Une clique de  $G$

$$V' = \{v_1, v_2, v_3\}$$

$$E' = \{e_1, e_2, e_3\}$$

### Exercice 2.1

Montrez que dans un groupe de six personnes, il y en a nécessairement trois qui se connaissent mutuellement ou trois qui ne se connaissent pas (on suppose que si A connaît B, B connaît également A).

Montrez que cela n'est plus nécessairement vrai dans un groupe de cinq personnes.

## 3. Degrés

### Degré d'un sommet

Pour un graphe ou un multigraphe, on appelle **degré du sommet**  $v$ , et on note  $d(v)$ , le nombre d'arêtes incidentes avec ce sommet. **Attention !** une boucle sur un sommet est comptée deux fois.

Dans un graphe simple, on peut aussi définir le degré d'un sommet comme étant le nombre de ses voisins (la taille de son voisinage).

Dans le graphe ci-contre, on a les degrés :

$$d(v_1) = 2$$

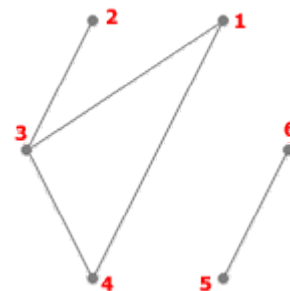
$$d(v_2) = 1$$

$$d(v_3) = 3$$

$$d(v_4) = 2$$

$$d(v_5) = 1$$

$$d(v_6) = 1$$



### Lemme des poignées de mains

La somme des degrés des sommets d'un graphe est égale à deux fois le nombre d'arêtes.

### Degré d'un graphe

Le **degré d'un graphe** est le degré maximum de tous ses sommets. Dans l'exemple ci-dessus, le degré du graphe est 3.

Un graphe dont tous les sommets ont le même degré est dit **régulier**. Si le degré commun est  $k$ , alors on dit que le graphe est  **$k$ -régulier**.

### Exercice 3.1

Une suite décroissante (au sens large) d'entiers est graphique s'il existe un graphe simple dont les degrés des sommets correspondent à cette suite (par exemple, le triangle à trois sommets correspond à la suite 2, 2, 2). Les suites suivantes sont-elles graphiques ?

- a) 3, 3, 2, 1, 1
- b) 3, 3, 1, 1
- c) 3, 3, 2, 2
- d) 4, 2, 1, 1, 1, 1
- e) 5, 3, 2, 1, 1, 1
- f) 5, 4, 3, 1, 1, 1, 1

Trouvez deux graphes distincts correspondant à la suite (3, 2, 2, 2, 1).

### Exercice 3.2

Démontrez le lemme des poignées de mains.

### Exercice 3.3

Montrez qu'un graphe simple a un nombre pair de sommets de degré impair.

### Exercice 3.4

Est-il possible de relier 15 ordinateurs de sorte que chaque appareil soit relié avec exactement trois autres ?

### Exercice 3.5

On s'intéresse aux graphes 3-réguliers. Construisez de tels graphes ayant 4, 5, 6, puis 7 sommets.

Qu'en déduisez-vous ? Prouvez-le !

### Exercice 3.6

Un groupe de 15 fans d'un chanteur célèbre, possède les deux particularités suivantes :

- chaque personne connaît au moins 7 autres
- toute information détenue par une personne est répercutée dans la minute qui suit à ses connaissances (et uniquement à elles)

Quel est le temps maximal entre le moment où un des 15 fans apprend une chose nouvelle sur leur idole, et celui où le groupe entier est au courant ?

---

## 4. Chaînes et cycles

Une **chaîne** dans  $G$ , est une suite de la forme  $(v_0, e_1, v_1, e_2, \dots, v_{k-1}, e_k, v_k)$  ayant pour éléments alternativement des sommets ( $v_i$ ) et des arêtes ( $e_i$ ), commençant et se terminant par un sommet, et telle que les extrémités de  $e_i$  soient  $v_{i-1}$  et  $v_i$ ,  $i = 1, \dots, k$ .

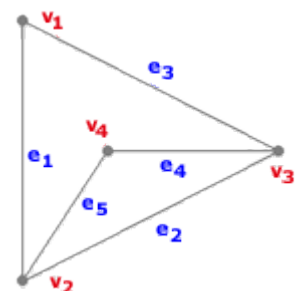
Si  $v_0 = a$  et  $v_k = b$ , on dira que la chaîne **relie**  $a$  et  $b$ . En plus, on dira que la chaîne a une **longueur**  $k$  (c'est le nombre d'arêtes de la chaîne). Une chaîne doit comporter au moins une arête.

Le graphe ci-contre contient les chaînes  $(v_1, e_3, v_3, e_4, v_4)$  et  $(v_4, e_4, v_3, e_2, v_2, e_1, v_1)$ , entre autres.

On ne change pas une chaîne en inversant l'ordre des éléments dans la suite correspondante, ainsi les chaînes  $(v_1, e_3, v_3, e_4, v_4)$  et  $(v_4, e_4, v_3, e_3, v_1)$  sont identiques.

On appelle **distance entre deux sommets** la longueur de la plus petite chaîne les reliant.

On appelle **diamètre d'un graphe** la plus longue des distances entre deux sommets.





Une chaîne est **élémentaire** si chaque sommet y apparaît au plus une fois.

Une chaîne est **simple** si chaque arête apparaît au plus une fois. Dans le graphe ci-dessus,  $(v_1, e_1, v_2, e_2, v_3)$  est une chaîne simple et élémentaire.

Une chaîne telle que  $v_0 = v_k$  est appelée **chaîne fermée**. Dans le graphe ci-dessus,  $(v_2, e_2, v_3, e_4, v_4, e_4, v_3, e_2, v_2)$  est une chaîne fermée.

Une chaîne fermée simple est appelée **cycle** si seul le sommet de départ apparaît deux fois dans la chaîne. Dans le graphe ci-dessus,  $(v_2, e_2, v_3, e_4, v_4, e_5, v_2)$  est un cycle.

## Théorème 1

Pour tout graphe ayant  $m$  arêtes,  $n$  sommets et  $p$  composantes connexes, on a :

$$v(G) = m - n + p \geq 0$$

De plus,  $v(G) = 0$  si et seulement si  $G$  est sans cycles.

$v(G)$  est appelé le **nombre cyclomatique**. Prononcer « nu de  $G$  ».

### Exercice 4.1

Dans certains livres, on définit une chaîne comme une suite de sommets. Dites pourquoi cette définition n'est pas tout à fait correcte.

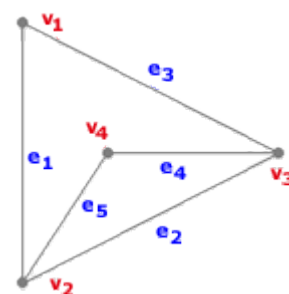
### Exercice 4.2

Démontrez le théorème 1.

### Exercice 4.3

Quel est le diamètre du graphe ci-contre ?

Combien d'arêtes (et lesquelles) faut-il enlever pour que son diamètre soit de 3 ?



### Exercice 4.4

Quels sont les graphes de diamètre 1 ?

## 5. Graphes eulériens

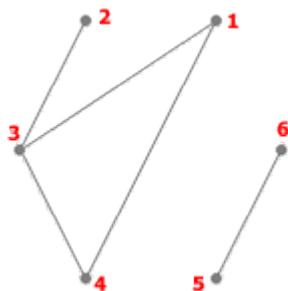
On dit qu'un graphe est **eulérien** s'il est possible de trouver un cycle passant une et une seule fois par toutes les arêtes.

On dit qu'un graphe est **semi-eulérien** s'il est possible de trouver une chaîne passant une et une seule fois par toutes les arêtes. Plus simplement, on peut dire qu'un graphe est eulérien (ou semi-eulérien) s'il est possible de dessiner le graphe sans lever le crayon (et sans passer deux fois sur le même trait).

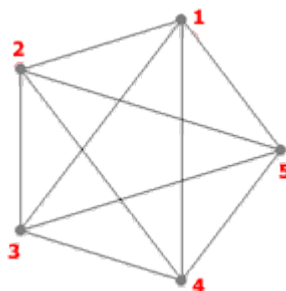
### Exercice 5.1

Les graphes suivants sont-ils eulériens (ou semi-eulériens) ?

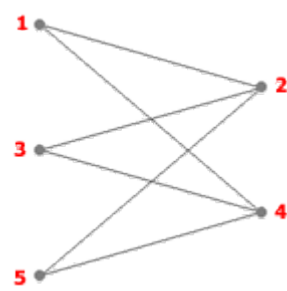
a.



b.



c.



### Exercice 5.2

Cet exercice est un des problèmes fondateurs de la théorie des graphes, proposé par le mathématicien suisse *Leonhard Euler* en 1736.

La ville de Königsberg (aujourd'hui Kaliningrad) est traversée par la Pregel, qui coule de part et d'autre de l'île de Kneiphof, et possède sept ponts (en rouge sur la figure ci-dessous):



Leonhard Euler  
(1707-1783)

Lors d'une promenade, est-il possible de passer sur tous les ponts de la ville une et une seule fois ?

### Exercice 5.3

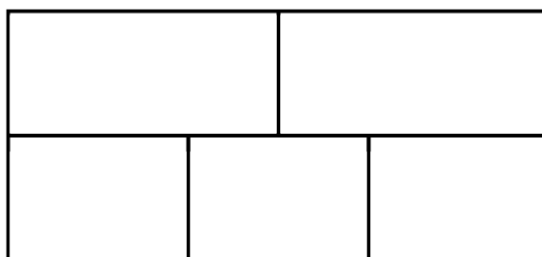
Donnez un critère permettant de dire à coup sûr si un graphe est eulérien.

### Exercice 5.4

Soit  $G$  un graphe non eulérien. Est-il toujours possible de rendre  $G$  eulérien en lui rajoutant un sommet et quelques arêtes ?

### Exercice 5.5

Est-il possible de tracer une courbe, sans lever le crayon, qui coupe chacun des 16 segments de la figure suivante exactement une fois ?



### Exercice 5.6

On considère des dominos dont les faces sont numérotées 1, 2, 3, 4 ou 5.

1. En excluant les dominos doubles, de combien de dominos dispose-t-on ?
2. Montrez que l'on peut arranger ces dominos de façon à former une boucle fermée (en utilisant la règle habituelle de contact entre les dominos).
3. Pourquoi n'est-il pas nécessaire de considérer les dominos doubles ?
4. Si l'on prend maintenant des dominos dont les faces sont numérotées de 1 à  $n$ , est-il possible de les arranger de façon à former une boucle fermée ?

## 6. Graphes hamiltoniens

On dit qu'un graphe est **hamiltonien** s'il est possible de trouver un cycle passant une et une seule fois par tous les sommets.

On dit qu'un graphe est **semi-hamiltonien** s'il est possible de trouver une chaîne passant une et une seule fois par tous les sommets.

Contrairement aux graphes eulériens, il n'existe pas de caractérisation simple des graphes hamiltoniens ou semi-hamiltoniens. On peut cependant énoncer quelques propriétés et conditions suffisantes.

- Un graphe possédant un sommet de degré 1 ne peut être hamiltonien.
- Si un sommet dans un graphe est de degré 2, alors les deux arêtes incidentes à ce sommet doivent faire partie du cycle hamiltonien.
- Les graphes complets  $K_n$  sont hamiltoniens.

### Théorème 2 (Ore)

Soit  $G = (V, E)$  un graphe simple d'ordre  $n \geq 3$ .

Si pour toute paire  $\{x, y\}$  de sommets non adjacents, on a  $d(x) + d(y) \geq n$ , alors  $G$  est hamiltonien.

### Corollaire (Dirac)

Soit  $G = (V, E)$  un graphe simple d'ordre  $n \geq 3$ . Si pour tout sommet  $x$  de  $G$ , on a  $d(x) \geq n/2$ , alors  $G$  est hamiltonien.

En effet, un tel graphe vérifie les conditions du théorème précédent. Si  $x$  et  $y$  ne sont pas adjacents, on a bien :

$$d(x) + d(y) \geq n/2 + n/2 = n$$

### Problème du voyageur de commerce

Le problème du voyageur de commerce (Traveling Salesman Problem en anglais) consiste à trouver le **plus court cycle passant par tous les sommets** dans un graphe où les arêtes sont pondérées. On travaille souvent sur un graphe complet.

#### Exercice 6.1

Dessinez un graphe d'ordre au moins 5 qui est...

1. hamiltonien et eulérien
2. hamiltonien et non eulérien
3. non hamiltonien et eulérien
4. non hamiltonien et non eulérien.

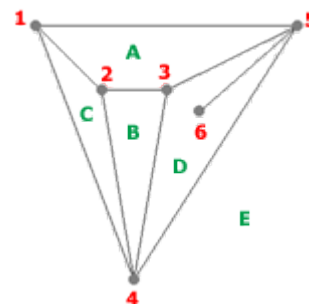
## 7. Graphes planaires

On dit qu'un graphe est **planaire** si on peut le dessiner dans le plan de sorte que ses arêtes ne se croisent pas. Rappelons que les arêtes ne sont pas forcément rectilignes.

Une **carte**, ou **graphe planaire topologique**, est une représentation particulière d'un multigraphe planaire fini. On dit qu'une carte est **connexe** si son graphe l'est. Une carte divise le plan en plusieurs **régions**.

Par exemple, la carte ci-contre, avec six sommets et neuf arêtes, divise le plan en cinq régions (A, B, C, D, E). On remarque que quatre régions sont limitées alors que la cinquième (E), extérieure au diagramme, ne l'est pas.

Le **degré d'une région**  $r$ , noté  $d(r)$ , est la longueur du cycle ou de la chaîne fermée qui limite  $r$ . Dans le graphe ci-contre,  $d(A)=4$ ,  $d(B)=3$ ,  $d(C)=3$ ,  $d(D)=5$ ,  $d(E)=3$ .



On remarque que toute arête limite deux régions, ou est contenue dans une région et est alors comptée deux fois dans la chaîne fermée. Nous avons donc un lemme pour les régions, analogue au lemme des poignées de mains pour les sommets.

## Lemme

La somme des degrés des régions d'une carte connexe est égale à deux fois le nombre d'arêtes.

On peut vérifier ce lemme sur le graphe ci-dessus : il comporte 9 arêtes et la somme des degrés des régions vaut 18.

## Théorème 3 (Euler)

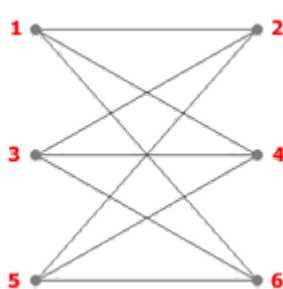
Euler a établi une formule célèbre qui relie le nombre de sommets  $S$ , le nombre d'arêtes  $A$  et le nombre de régions  $R$  d'une carte connexe.

$$S - A + R = 2$$

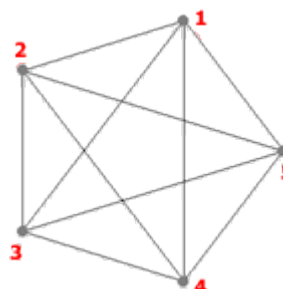
Pendant de nombreuses années, les mathématiciens ont tenté de caractériser les graphes planaires. Ce problème a été résolu en 1930 par le mathématicien polonais K. Kuratowski. La démonstration du théorème sort du cadre de ce cours.

## Théorème 4 (Kuratowski)

Un graphe est non planaire si et seulement si il contient un sous-graphe homéomorphe à  $K_{3,3}$  ou  $K_5$ .



$K_{3,3}$



$K_5$

### Exercice 7.1

Démontrez le théorème d'Euler. Procédez par induction.

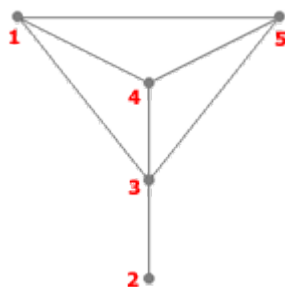
### Exercice 7.2

Utilisez le théorème d'Euler pour démontrer que le graphe  $K_{3,3}$  n'est pas planaire.

## 8. Matrice et listes d'adjacences

### Matrice d'adjacences

On peut représenter un graphe par une matrice d'adjacences. Une matrice  $(n \times m)$  est un tableau de  $n$  lignes et  $m$  colonnes.  $(i, j)$  désigne l'intersection de la ligne  $i$  et de la colonne  $j$ . Dans une matrice d'adjacences, les lignes et les colonnes représentent les sommets du graphe. Un 1 à la position  $(i, j)$  signifie que le sommet  $i$  est adjacent au sommet  $j$ .



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

Cette matrice a plusieurs caractéristiques :

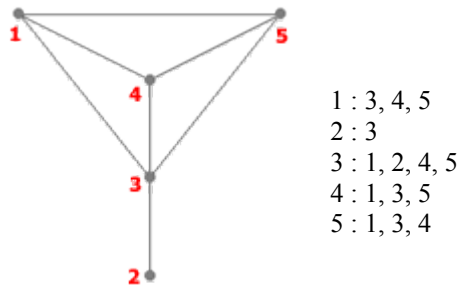
1. Elle est carrée : il y a autant de lignes que de colonnes.
2. Il n'y a que des zéros sur la diagonale. Un 1 sur la diagonale indiquerait une boucle.
3. Elle est symétrique :  $m_{ij} = m_{ji}$ .

Les matrices d'adjacences ont des propriétés intéressantes. On a calculé ci-dessous les matrices  $M^2$  et  $M^3$ . Pour chacune de ces matrices, à quoi correspondent les nombres obtenus ?

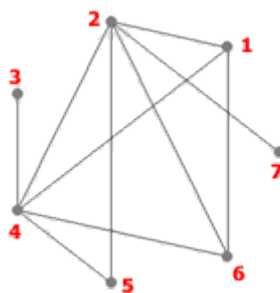
$$M^2 = \begin{pmatrix} 3 & 1 & 2 & 2 & 2 \\ 1 & 1 & 0 & 1 & 1 \\ 2 & 0 & 4 & 2 & 2 \\ 2 & 1 & 2 & 3 & 2 \\ 2 & 1 & 2 & 2 & 3 \end{pmatrix} \quad M^3 = \begin{pmatrix} 6 & 2 & 8 & 7 & 7 \\ 2 & 0 & 4 & 2 & 2 \\ 8 & 4 & 6 & 8 & 8 \\ 7 & 2 & 8 & 6 & 7 \\ 7 & 2 & 8 & 7 & 6 \end{pmatrix}$$

## Listes d'adjacences

On peut aussi représenter un graphe en donnant pour chacun de ses sommets la liste des sommets auxquels il est adjacent.



### Exercice 8.1

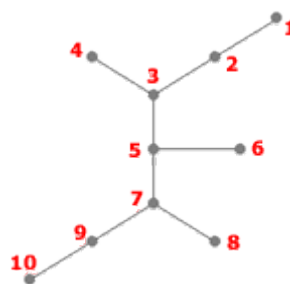


Décrivez le graphe  $G$  ci-contre par une matrice d'adjacences et des listes d'adjacences.

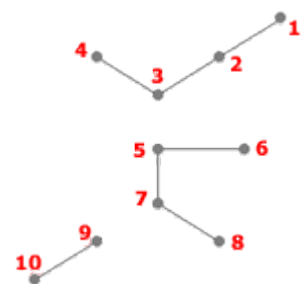
## 9. Arbres

On appelle **arbre** tout graphe connexe, sans cycles.

Un graphe sans cycles mais non connexe est appelé une **forêt**.



Un arbre



Une forêt

### Théorème 5

Les affirmations suivantes sont équivalentes pour tout graphe  $G = (V, E)$  à  $n$  sommets.

1.  $G$  est un arbre,
2.  $G$  est sans cycles et connexe,
3.  $G$  est sans cycles et comporte  $n-1$  arêtes,
4.  $G$  est connexe et comporte  $n-1$  arêtes,
5. chaque paire  $\{u, v\}$  de sommets distincts est reliée par une seule chaîne simple (et le graphe est sans boucles).

## Théorème 6

Tout arbre fini avec au moins deux sommets comporte au moins deux sommets pendants ou feuilles, c'est-à-dire des sommets incidents avec une seule arête.

### Exercice 9.1

Démontrez le théorème 5. Pour cela, utilisez le théorème 1.

### Exercice 9.2

Démontrez le théorème 6.

### Exercice 9.3

Combien d'arbres différents existe-t-il avec 5 sommets ? avec 6 sommets ? avec 7 sommets ?

## 10. Codage de Prüfer

Le codage de Prüfer est une manière très compacte de décrire un arbre.

Soit l'arbre  $T = (V, E)$  et supposons  $V = \{1, 2, \dots, n\}$ .

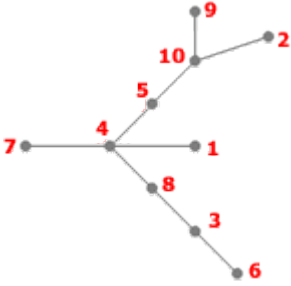
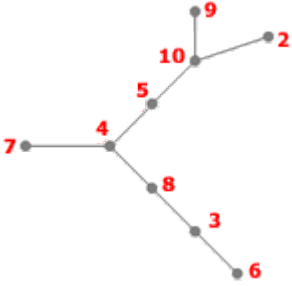
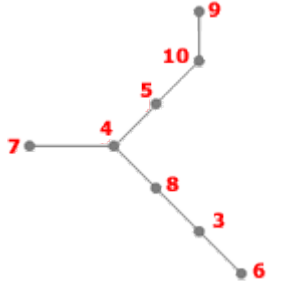
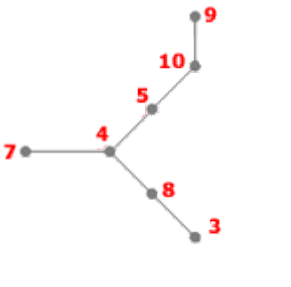
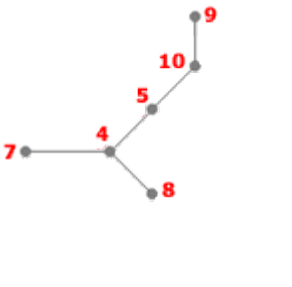
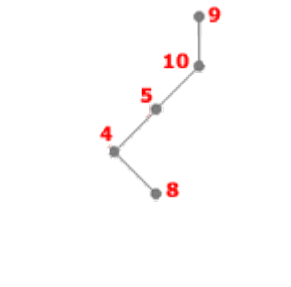
L'algorithme ci-dessous fournira le code de  $T$ , c'est-à-dire une suite  $S$  de  $n-2$  termes employant (éventuellement plusieurs fois) des nombres choisis parmi  $\{1, \dots, n\}$ .


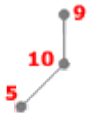

### Pas général de l'algorithme de codage

(à répéter tant qu'il reste plus de deux sommets dans l'arbre courant  $T$ )

- identifier la feuille  $v$  de l'arbre courant ayant le numéro minimum ;
- ajouter à la suite  $S$  le seul sommet  $s$  adjacent à  $v$  dans l'arbre  $T$  courant ;
- enlever de l'arbre  $T$  courant le sommet  $v$  et l'arête incidente à  $v$ .

### Exemple de codage

 <p><b>Etape 0</b> Arbre à coder</p>	 <p><b>Etape 1</b> <math>S = \{4\}</math></p>	 <p><b>Etape 2</b> <math>S = \{4, 10\}</math></p>
 <p><b>Etape 3</b> <math>S = \{4, 10, 3\}</math></p>	 <p><b>Etape 4</b> <math>S = \{4, 10, 3, 8\}</math></p>	 <p><b>Etape 5</b> <math>S = \{4, 10, 3, 8, 4\}</math></p>

 <p><b>Etape 6</b>  <math>S = \{4, 10, 3, 8, 4, 4\}</math></p>	 <p><b>Etape 7</b>  <math>S = \{4, 10, 3, 8, 4, 4, 5\}</math></p>	 <p><b>Etape 8</b>  <math>S = \{4, 10, 3, 8, 4, 4, 5, 10\}</math></p>
---	--	--

$S = \{4, 10, 3, 8, 4, 4, 5, 10\}$  est le codage de Prüfer de l'arbre initial.

## Décodage

Donnée : suite  $S$  de  $n-2$  nombres, chacun provenant de  $\{1, \dots, n\}$ .

Posons  $I = \{1, \dots, n\}$ .

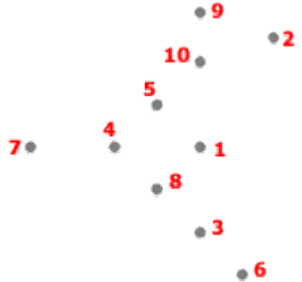
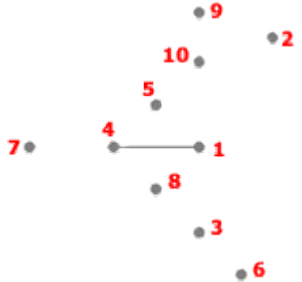
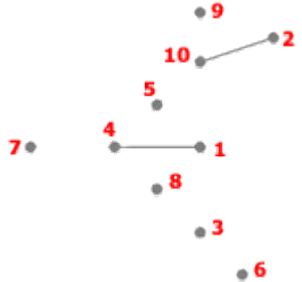
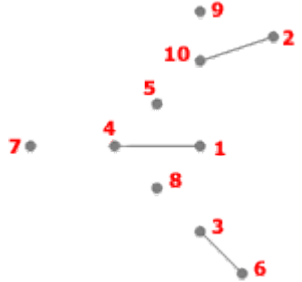
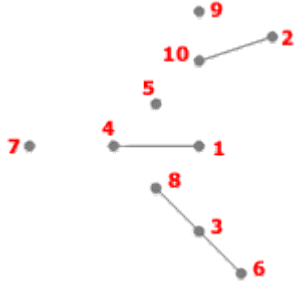
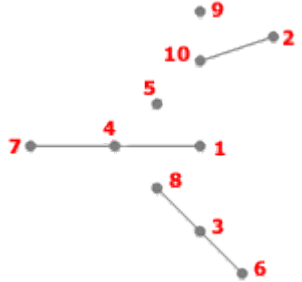
### Pas général de l'algorithme de décodage

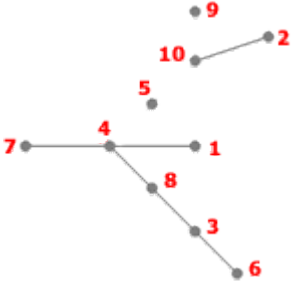
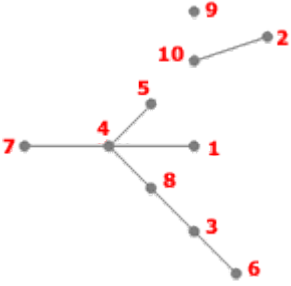
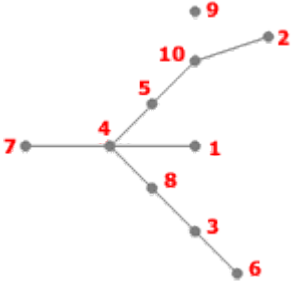
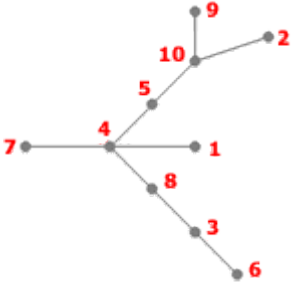
(à répéter tant qu'il reste des éléments dans  $S$  et plus de deux éléments dans  $I$ )

- identifier le plus petit élément  $i$  de  $I$  n'apparaissant pas dans la suite  $S$  ;
- relier par une arête de  $T$  le sommet  $i$  avec le sommet  $s$  correspondant au premier élément de la suite  $S$  ;
- enlever  $i$  de  $I$  et  $s$  de  $S$ .

Les deux éléments qui restent dans  $I$  à la fin de l'algorithme constituent les extrémités de la dernière arête à ajouter à  $T$ .

### Exemple de décodage

 <p><b>Etape 1</b>  <math>I = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}</math>  <math>S = \{4, 10, 3, 8, 4, 4, 5, 10\}</math></p>	 <p><b>Etape 2</b>  <math>I = \{2, 3, 4, 5, 6, 7, 8, 9, 10\}</math>  <math>S = \{10, 3, 8, 4, 4, 5, 10\}</math></p>	 <p><b>Etape 3</b>  <math>I = \{3, 4, 5, 6, 7, 8, 9, 10\}</math>  <math>S = \{3, 8, 4, 4, 5, 10\}</math></p>
 <p><b>Etape 4</b>  <math>I = \{3, 4, 5, 7, 8, 9, 10\}</math>  <math>S = \{8, 4, 4, 5, 10\}</math></p>	 <p><b>Etape 5</b>  <math>I = \{4, 5, 7, 8, 9, 10\}</math>  <math>S = \{4, 4, 5, 10\}</math></p>	 <p><b>Etape 6</b>  <math>I = \{4, 5, 8, 9, 10\}</math>  <math>S = \{4, 5, 10\}</math></p>

 <p><b>Etape 7</b></p> <p><math>I = \{4, 5, 9, 10\}</math>  <math>S = \{5, 10\}</math></p>	 <p><b>Etape 8</b></p> <p><math>I = \{5, 9, 10\}</math>  <math>S = \{10\}</math></p>	 <p><b>Etape 9</b></p> <p><math>I = \{9, 10\}</math>  <math>S = \{\}</math></p>
 <p><b>Etape 10</b></p> <p><math>I = \{\}</math>  <math>S = \{\}</math></p>		

### Théorème 7 (Cayley, 1857)

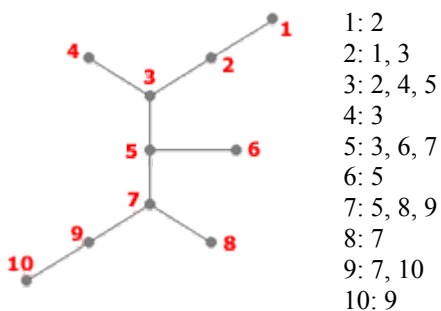
Le nombre d'arbres que l'on peut construire sur  $n$  ( $n > 1$ ) sommets numérotés est égal à  $n^{n-2}$ .

### Preuve

La preuve est immédiate en utilisant le codage de Prüfer. En effet, on vérifie aisément que les deux algorithmes constituent les deux sens d'une bijection entre les arbres sur  $n$  sommets numérotés et les mots de  $n-2$  lettres sur l'alphabet à  $n$  lettres.

Ce constat permet de conclure la preuve du théorème de Cayley. En effet, il existe  $n^{n-2}$  mots de longueur  $n-2$  sur l'alphabet à  $n$  lettres, donc d'arbres sur  $n$  sommets numérotés.

### Exercice 10.1



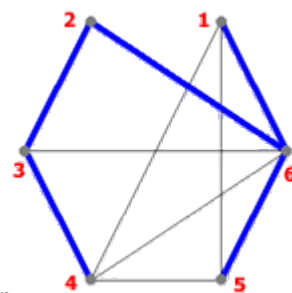
Décrivez l'arbre ci-contre à l'aide du codage de Prüfer.

Dessinez l'arbre correspondant à la suite  $S = \{1, 1, 1, 1, 1, 1, 1, 1\}$ .



# 11. Arbres couvrants

Un **arbre couvrant** (ou arbre maximal) est un graphe partiel qui est aussi un arbre. Ci-contre, un des arbres couvrants (en bleu) d'un graphe donné.



## Arbre maximal de poids minimum

On considérera le problème suivant :

Soit le graphe  $G = (V, E)$  avec un poids associé à chacune de ses arêtes. Trouver, dans  $G$ , un arbre maximal  $A = (V, F)$  de poids total minimum.

### L'algorithme de Kruskal

**Données :** Graphe  $G = (V, E)$  ( $|V| = n$ ,  $|E| = m$ ) et pour chaque arête  $e$  de  $E$ , son poids  $c(e)$ .

**Résultat :** Arbre ou forêt maximale  $A = (V, F)$  de poids minimum.

Trier et renuméroter les arêtes de  $G$  dans l'ordre croissant de leur poids :  $c(e_1) \leq c(e_2) \leq \dots \leq c(e_m)$ .

Poser  $F := \emptyset$ ,  $k := 0$

**Tant que**  $k < m$  et  $|F| < n-1$  **faire**

**Début**

    si  $e_{k+1}$  ne forme pas de cycle avec  $F$  alors  $F := F \cup \{e_{k+1}\}$

$k := k+1$

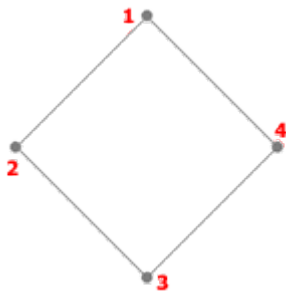
**Fin**

Ce problème de trouver un arbre maximal de poids minimum se pose, par exemple, lorsqu'on désire relier  $n$  villes par un réseau routier de coût minimum. Les sommets du graphe représentent les villes, les arêtes, les tronçons qu'il est possible de construire et les poids des arêtes correspondent aux coûts de construction du tronçon correspondant. L'algorithme de Kruskal décrit ci-dessus permet de résoudre ce problème.

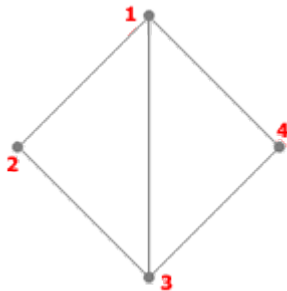
### Exercice 11.1

Combien d'arbres couvrants différents les graphes suivants possèdent-ils ?

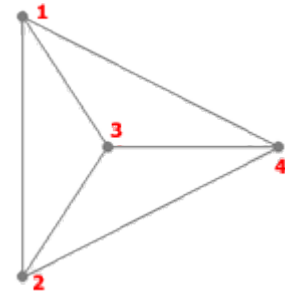
a.



b.

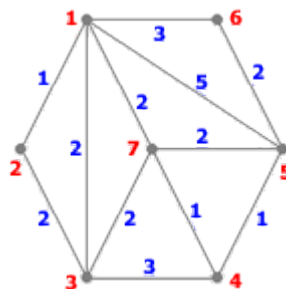


c.



### Exercice 11.2

Trouvez tous les arbres maximaux de poids minimum du graphe ci-contre (les chiffres en bleu représentent le poids des arêtes) :



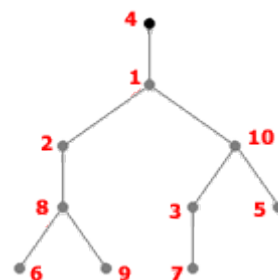
## 12. Arborescences



On appelle **arborescence** un arbre avec un sommet distingué, que l'on appelle la racine. On représente généralement une arborescence avec la racine en haut du dessin et les feuilles en bas.

Sur l'arborescence ci-contre, la racine est le sommet 4. Les sommets 5, 6, 7 et 9 sont les feuilles.

On peut, dans une arborescence, assigner un **rang** aux sommets. Le rang sera en fait la distance du sommet à la racine. Ainsi, dans l'exemple ci-contre, le sommet 4 (la racine) a rang 0, le sommet 1 a rang 1, les sommets 2 et 10 ont rang 2, les sommets 3, 5 et 8 ont rang 3 et les sommets 6, 7 et 9 ont rang 4.

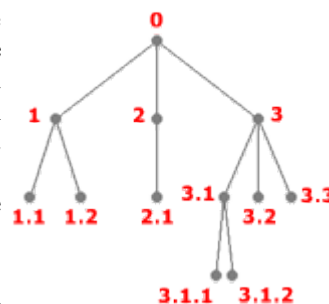


Une arborescence

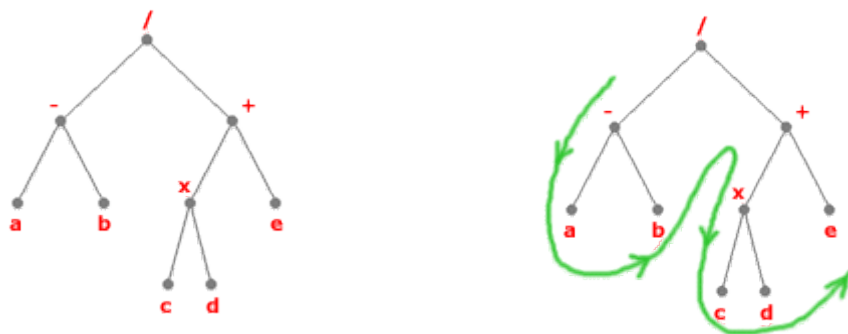
On dira que la **hauteur** de l'arborescence est le rang maximum (4 dans l'exemple ci-contre).

### Arborescences ordonnées

Une **arborescence ordonnée** est une arborescence dont les arêtes partant de chaque sommet sont ordonnées. On peut systématiquement étiqueter les sommets d'un tel arbre comme suit : on attribue 0 à la racine  $r$ , puis 1, 2, 3, ... aux sommets qui sont adjacents à  $r$ . Les étiquettes suivantes sont constituées de l'étiquette du sommet « père », suivie d'un chiffre obtenu comme précédemment. Ainsi, les sommets « fils » attachés au sommet 2 seront numérotés 2.1, 2.2, 2.3, ... La figure ci-contre illustre le procédé. Cet ordre (0, 1, 1.1, 1.2, 2, 2.1, 3, 3.1, 3.1.1, 3.1.2, 3.2, 3.3) est appelé **ordre lexicographique**, puisqu'il est semblable au classement des mots dans un dictionnaire. Il est identique à l'ordre obtenu en parcourant de haut en bas la branche la plus à gauche de l'arbre, puis la branche immédiatement à droite, et ainsi de suite jusqu'à la branche la plus à droite. On parle aussi de **parcours en profondeur** de l'arbre, par opposition au **parcours en largeur** qui serait l'ordre: 0, 1, 2, 3, 1.1, 1.2, 2.1, 3.1, 3.2, 3.3, 3.1.1, 3.1.2.



N'importe quelle expression algébrique comprenant des expressions binaires, telle que l'addition, la soustraction, la multiplication et la division, peut être représentée par une arborescence ordonnée. Par exemple, l'arborescence ci-dessous représente l'expression arithmétique  $(a - b) / ((c \times d) + e)$  :



On observe que les variables de l'expression ( $a$ ,  $b$ ,  $c$ ,  $d$  et  $e$ ) sont les feuilles et que les opérations sont les autres sommets. L'arbre doit être ordonné car  $a - b$  et  $b - a$  conduisent au même arbre, mais pas au même arbre ordonné. Le mathématicien polonais *Lukasiewicz* a remarqué qu'en plaçant les symboles des opérations binaires avant les arguments, c'est-à-dire  $+ ab$  au lieu de  $a + b$  ou  $/ cd$  au lieu de  $c / d$ , nous n'avons plus besoin de parenthèses. Il s'agit de la **notation polonaise** dans sa forme préfixée ou **directe** (par analogie, en notation polonaise postfixée ou inverse, on place les symboles après les arguments ; certaines calculatrices, notamment les HP, utilisent la notation polonaise inverse). L'expression  $(a - b) / ((c \times d) + e)$  devient ainsi  $/ - a b + \times c d e$  (voir le chemin vert de l'arborescence ci-dessus).

### Exercice 12.1

Combien d'arborescences existe-t-il sur  $n$  sommets numérotés ?

### Exercice 12.2

Étant donnée l'expression algébrique  $(2x + y) \cdot (5a - b)^3$ ,

- dessinez l'arborescence ordonnée correspondante ;
- trouvez la portée de l'exponentiation (la **portée** d'un sommet  $s$  dans une arborescence est le sous-arbre généré par  $s$  et les sommets qui le suivent avec  $s$  pour racine) ;
- écrivez l'expression en notation polonaise directe.

---

## 13. Codage de Huffman

Le **codage de Huffman** est une méthode de **compression statistique de données** qui permet de réduire la longueur du codage d'un alphabet. Le code de Huffman (1952) est un code de longueur variable optimal, c'est-à-dire tel que la longueur moyenne d'un texte codé soit minimale. On observe ainsi des réductions de taille de l'ordre de 20 à 90%.

### Le principe

Le principe de l'algorithme de Huffman consiste à recoder les octets (suite de 8 bits) rencontrés dans un ensemble de données source avec des valeurs de longueur binaire variable.

L'unité de traitement est ramenée au bit. Huffman propose de **recoder les données qui ont une occurrence très faible sur une longueur binaire supérieure à la moyenne, et recoder les données très fréquentes sur une longueur binaire très courte**.

Ainsi, pour les données rares, nous perdons quelques bits regagnés pour les données répétitives. Par exemple, dans un fichier ASCII le « w » apparaissant 10 fois aura un code très long : 0101000001000. Ici la perte est de 40 bits ( $10 \times 4$  bits), car sans compression, il serait codé sur 8 bits au lieu de 12. Par contre, le caractère le plus fréquent comme le « e » avec 200 apparitions sera codé par 1. Le gain sera de 1400 bits ( $7 \times 200$  bits). On voit l'intérêt d'une telle méthode.

De plus, le codage de Huffman a une **propriété de préfixe : une séquence binaire ne peut jamais être à la fois représentative d'un élément codé et constituer le début du code d'un autre élément**.

Si un caractère est représenté par la combinaison binaire 100 alors la combinaison 10001 ne peut être le code d'aucune autre information. Dans ce cas, l'algorithme de décodage interpréterait les 5 bits comme une succession du caractère codé 100 puis du caractère codé 01. Cette caractéristique du codage de Huffman permet une codification à l'aide d'une structure d'arbre binaire.

### Méthode

L'algorithme opère sur une forêt. Une forêt est ici un ensemble d'arbres étiquetés complets : tout noeud interne (c'est-à-dire qui n'est pas une feuille) a deux fils non-vides.

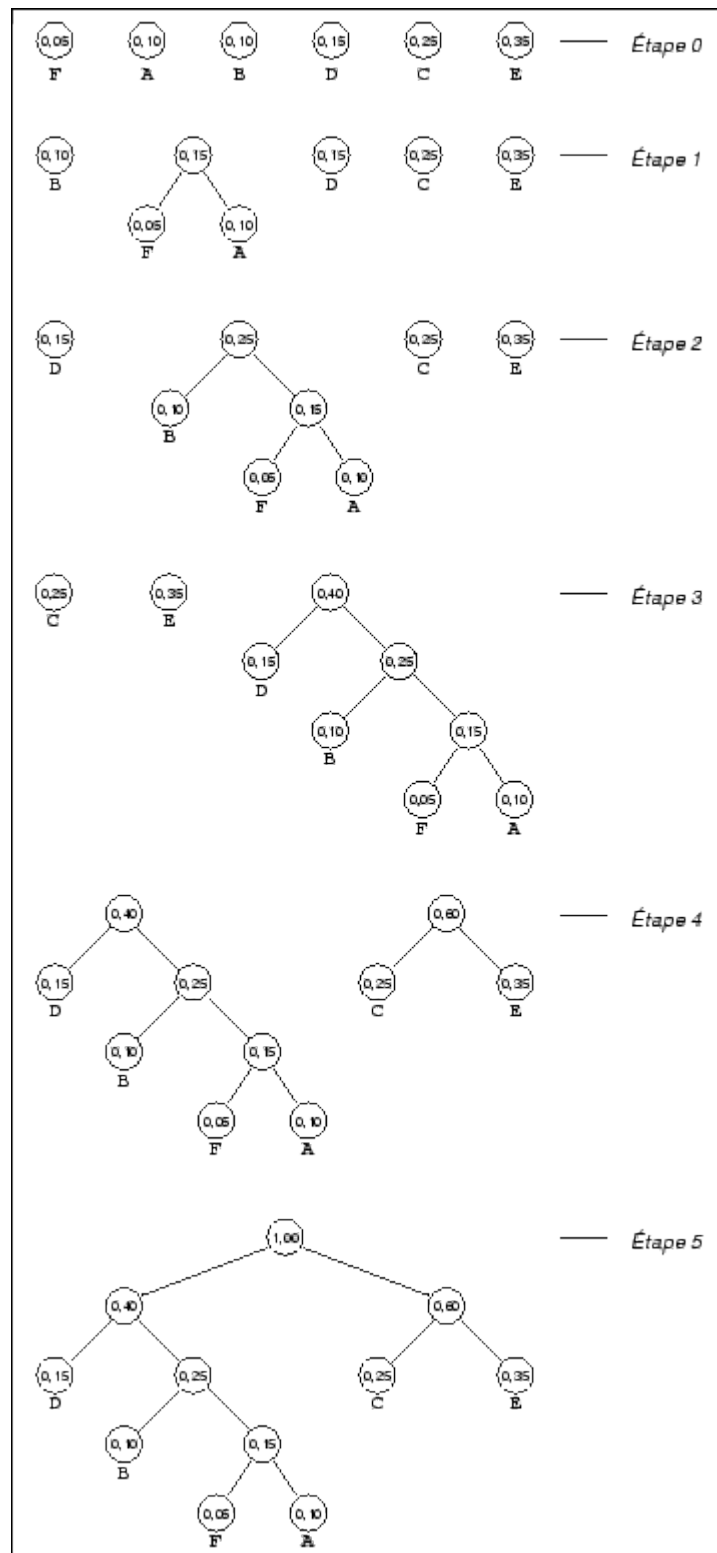
La forêt initiale est formée d'un arbre à un noeud pour chaque lettre du langage-source, dont l'étiquette est la probabilité de cette lettre. La forêt finale est formée d'un unique arbre, qui est l'arbre de décodage du code.

L'algorithme est de type glouton : il choisit à chaque étape les deux arbres d'étiquettes minimales, soit  $x$  et  $y$ , et les remplace par l'arbre formé de  $x$  et  $y$  et ayant comme étiquette la somme de l'étiquette de  $x$  et de l'étiquette de  $y$ .

La figure ci-dessous représente les étapes de la construction d'un code de Huffman pour l'alphabet source  $\{A, B, C, D, E, F\}$ , avec les probabilités  $P(A)=0.10$ ,  $P(B)=0.10$ ,  $P(C)=0.25$ ,  $P(D)=0.15$ ,  $P(E)=0.35$  et  $P(F)=0.05$ .

Le code d'une lettre est alors déterminé en suivant le chemin depuis la racine de l'arbre jusqu'à la feuille associée à cette lettre en concaténant successivement un 0 ou un 1 selon que la branche suivie est à gauche ou à droite. Ainsi, sur la figure ci-contre,  $A=0111$ ,  $B=010$ ,  $C=10$ ,  $D=00$ ,  $E=11$  et  $F=0110$ .

Par exemple le mot FADE serait codé 011001110011. Pour décoder, on lit simplement la chaîne de bits de gauche à droite. Le seul « découpage » possible est, grâce à la propriété du préfixe, 0110-0111-00-11.



## Conclusion

Ce principe de compression est aussi utilisé dans le codage d'image TIFF (Tagged Image Format File) spécifié par Microsoft Corporation et Aldus Corporation.

Par ailleurs, le codage d'image est fait en retranscrivant exactement le contenu d'un écran (image), en utilisant les méthodes traditionnelles de compression. Il existe des méthodes qui ne conservent pas exactement le contenu d'une image (méthodes non conservatives) mais dont la représentation visuelle reste correcte. Entre autres, il y a la méthode JPEG (Join Photographic Experts Group) qui utilise la compression de type Huffman pour coder les informations d'une image.

Malgré son ancienneté, cette méthode est toujours remise au goût du jour, et offre des performances appréciables. En effet, beaucoup de recherches en algorithmique ont permis d'améliorer les fonctionnalités de la méthode Huffman de base, par exemple les arbres binaires, les arbres équilibrés, etc.

### Exercice 13.1

Construisez un codage de Huffman du message "ceciestuncodagedehuffman" (on a supprimé les espaces et la ponctuation pour simplifier la construction). Il y a plusieurs codages de Huffman possibles. Vérifiez la propriété du préfixe.

### Exercice 13.2 (pour les plus courageux)

Utilisez le tableau ci-dessous pour déterminer le codage de Huffman de la langue française.

Fréquences d'apparition des lettres en français

Lettre	Fréquence	Lettre	Fréquence
A	8.40 %	N	7.13 %
B	1.06 %	O	5.26 %
C	3.03 %	P	3.01 %
D	4.18 %	Q	0.99 %
E	17.26 %	R	6.55 %
F	1.12 %	S	8.08 %
G	1.27 %	T	7.07 %
H	0.92 %	U	5.74 %
I	7.34 %	V	1.32 %
J	0.31 %	W	0.04 %
K	0.05 %	X	0.45 %
L	6.01 %	Y	0.30 %
M	2.96 %	Z	0.12 %

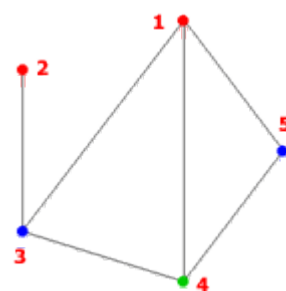
## 14. Coloration des sommets

Soit  $G = (V, E)$  un graphe. Un sous-ensemble  $S$  de  $V$  est un **stable** s'il ne comprend que des sommets non adjacents deux à deux. Le cardinal du plus grand stable est le **nombre de stabilité** de  $G$  ; on le note  $\alpha(G)$ .

La coloration des sommets d'un graphe consiste à affecter à tous les sommets de ce graphe une couleur de telle sorte que deux sommets adjacents ne portent pas la même couleur. Une coloration avec  $k$  couleurs est donc une partition de l'ensemble des sommets en  $k$  stables. Le **nombre chromatique** du graphe  $G$ , noté  $\chi(G)$ , est le plus petit entier  $k$  pour lequel il existe une partition de  $V$  en  $k$  sous-ensembles stables.

Sur le graphe ci-contre, on a eu besoin de trois couleurs pour colorer les sommets de sorte que deux sommets adjacents ont des couleurs différentes. On a donc trois stables :  $\{1, 2\}$ ,  $\{3, 5\}$  et  $\{4\}$ . On ne peut pas utiliser moins de couleurs, à cause des cliques 1-4-5 et 1-3-4.

Remarquons enfin que le sommet 2 aurait pu aussi être vert. La coloration minimale n'est donc pas forcément unique.



Une coloration des sommets

## Encadrement du nombre chromatique

### Majoration

1.  $\gamma(G) \leq r + 1$ , où  $r$  est le plus grand degré de ses sommets.

**Preuve :** Soit un graphe et  $r$  le degré maximum de ses sommets. Donnons-nous une palette de  $(r + 1)$  couleurs. Pour chaque sommet du graphe on peut tenir le raisonnement suivant : ce sommet est adjacent à  $r$  sommets au plus, et le nombre de couleurs déjà utilisées pour colorer ces sommets est donc inférieur ou égal à  $r$ . Il reste donc au moins une couleur non utilisée dans la palette, avec laquelle nous pouvons colorer notre sommet.

2.  $\gamma(G) \leq n + 1 - \alpha(G)$

**Preuve :** Considérons  $S$  un stable de  $V$  de cardinal  $\alpha(G)$ . Une coloration possible des sommets consiste à colorer les sommets de  $S$  d'une même couleur et les  $n - \alpha(G)$  autres sommets de couleurs toutes différentes. On en déduit que  $\gamma(G) \leq 1 + (n - \alpha(G))$ .

### Minoration

1. Le nombre chromatique d'un graphe est supérieur ou égal à celui de chacun de ses sous-graphes.

**Preuve :** Ce résultat découle de la définition même du nombre chromatique.

2.  $\gamma(G) \geq \omega(G)$

**Preuve :** Puisque, par définition, dans une clique d'ordre  $m$ , tous les sommets sont adjacents entre eux, il faudra  $m$  couleurs. Donc, forcément, le nombre chromatique du graphe sera supérieur ou égal à l'ordre de sa plus grande clique.

## Algorithme de coloration de Welsh et Powell

Cet algorithme couramment utilisé permet d'obtenir une assez bonne coloration d'un graphe, c'est-à-dire une coloration n'utilisant pas un trop grand nombre de couleurs. Cependant il n'assure pas que le nombre de couleurs utilisé soit minimum (et donc égal au nombre chromatique du graphe).

### Étape 1

Classer les sommets du graphe dans l'ordre décroissant de leur degré, et attribuer à chacun des sommets son numéro d'ordre dans la liste obtenue.

### Étape 2

En parcourant la liste dans l'ordre, attribuer une couleur non encore utilisée au premier sommet non encore coloré, et attribuer cette même couleur à chaque sommet non encore coloré et non adjacent à un sommet de cette couleur.

### Étape 3

S'il reste des sommets non colorés dans le graphe, revenir à l'étape 2. Sinon, la coloration est terminée.

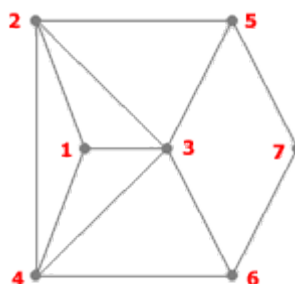
### Exercice 14.1

Tout graphe contenant un triangle ( $K_3$ ) ne peut pas être coloré en moins de trois couleurs.

1. Construire un graphe sans  $K_3$  qui nécessite également trois couleurs.
2. Comment, à partir du graphe précédent, construire un graphe sans  $K_4$  nécessitant 4 couleurs ?
3. Un graphe sans  $K_5$  nécessitant 5 couleurs ?

### Exercice 14.2

Déterminez le nombre chromatique de ce graphe.



### Exercice 14.3

Sept élèves, désignés par A, B, C, D, E, F et G, se sont rendus à la bibliothèque aujourd'hui. Le tableau suivant précise « qui a rencontré qui » (la bibliothèque étant petite, deux élèves présents au même moment se rencontrent nécessairement...).

l'élève	A	B	C	D	E	F	G
a rencontré	D,E	D,E,F,G	E,G	A,B,E	A,B,C,D,F,G	B,E,G	B,C,E,F

De combien de places assises doit disposer la bibliothèque pour que chacun ait pu travailler correctement au cours de cette journée ?

### Exercice 14.4

A, B, C, D, E, F, G et H désignent huit poissons. Dans le tableau ci-dessous, une croix signifie que les poissons ne peuvent pas cohabiter dans un même aquarium :

	A	B	C	D	E	F	G	H
A		x	x	x			x	x
B	x				x	x	x	
C	x			x		x	x	x
D	x		x		x			x
E		x		x		x	x	
F		x	x		x			
G	x	x	x		x			
H	x		x	x				

Quel nombre minimum d'aquariums faut-il ?

### Exercice 14.5

Un lycée doit organiser les horaires des examens. On suppose qu'il y a 7 épreuves à planifier, correspondant aux cours numérotés de 1 à 7 et que les paires de cours suivantes ont des étudiants communs: 1 et 2, 1 et 3, 1 et 4, 1 et 7, 2 et 3, 2 et 4, 2 et 5, 2 et 7, 3 et 4, 3 et 6, 3 et 7, 4 et 5, 4 et 6, 5 et 6, 5 et 7 et enfin 6 et 7.

Comment organiser ces épreuves de façon qu'aucun étudiant n'ait à passer deux épreuves en même temps et cela sur une durée minimale ?

### Exercice 14.6

Sept agences de voyage romaines proposent des visites de monuments et lieux touristiques : le Colisée, le Forum romain, le musée du Vatican et les thermes de Caracalas. Un même lieu ne peut être visité par plusieurs groupes de compagnies différentes le même jour.

La première Compagnie fait visiter uniquement le Colisée ; la seconde le Colisée et le musée du Vatican ; la troisième les thermes de Caracalas ; la quatrième le musée du Vatican et les thermes de Caracalas ; la cinquième le Colisée et le Forum romain ; la sixième le Forum romain et les thermes de Caracalas ; la septième le musée du Vatican et le forum romain.

Ces agences peuvent-elles organiser les visites sur les trois premiers jours de la semaine ?

### Exercice 14.7

Utilisez l'algorithme de coloration de Welsh et Powell pour colorer les graphes des exercices 14.2 et 14.5.

### Exercice 14.8

Exprimez la résolution d'un Sudoku classique en termes de coloration de graphe. Décrivez le graphe (nombre de sommets, nombre d'arêtes, etc.). Combien faut-il de couleurs ?

## 15. Coloration des sommets d'un graphe planaire

### Théorème 8 : théorème des quatre couleurs

On peut colorer les sommets d'un graphe planaire (sans boucles) en utilisant au plus quatre couleurs de telle sorte que toutes les arêtes aient des extrémités de couleurs différentes.

Cette conjecture a été formulée pour la première fois par l'Écossais *Francis Guthrie* en 1852. Il était alors question de coloration de carte de géographie (voir exercice 15.1). La preuve de ce théorème n'arriva qu'en... 1976, grâce à *Kenneth Appel* et *Wolfgang Haken*. La démonstration fit grand bruit car ce fut **le premier théorème de l'histoire des mathématiques qui a nécessité l'usage systématique de l'ordinateur**. La communauté mathématique se divisa alors en deux camps : ceux pour qui le théorème des quatre couleurs était définitivement démontré, et ceux pour qui tout restait à faire.

#### Exercice 15.1

Colorez la carte ci-dessous en utilisant le moins de couleurs possibles, de sorte que deux pays voisins aient des couleurs différentes (cliquez sur l'image pour avoir la carte de l'Afrique sans les couleurs).

Construisez d'abord un graphe associé à cette carte, puis colorez-en les sommets.



#### Exercice 15.2

Prenez une feuille de papier. Tracez une droite quelconque qui traverse la feuille de part en part. Recommencez l'opération  $n$  fois.

Démontrez que la « carte » ainsi obtenue peut être colorée en deux couleurs.



## 16. Coloration des arêtes

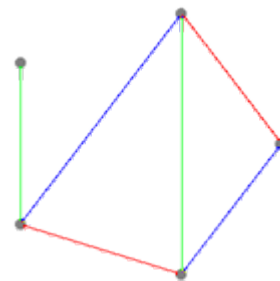
La coloration des arêtes d'un graphe consiste à affecter à toutes les arêtes de ce graphe une couleur de telle sorte que deux arêtes adjacentes ne portent pas la même couleur.

L'**indice chromatique** du graphe  $G$  est le plus petit entier  $k$  pour lequel il existe une coloration des arêtes; on le note  $\chi(G)$ .

Sur le graphe ci-contre, on a eu besoin de trois couleurs pour colorer les arêtes de sorte que deux arêtes adjacentes ont des couleurs différentes.

Pour colorer les arêtes d'un graphe, on peut se ramener au problème de la coloration des sommets. Il suffit pour cela de travailler non pas sur le graphe lui-même, mais sur le **graphe adjoint**, noté  $G'$ , et que l'on définit ainsi:

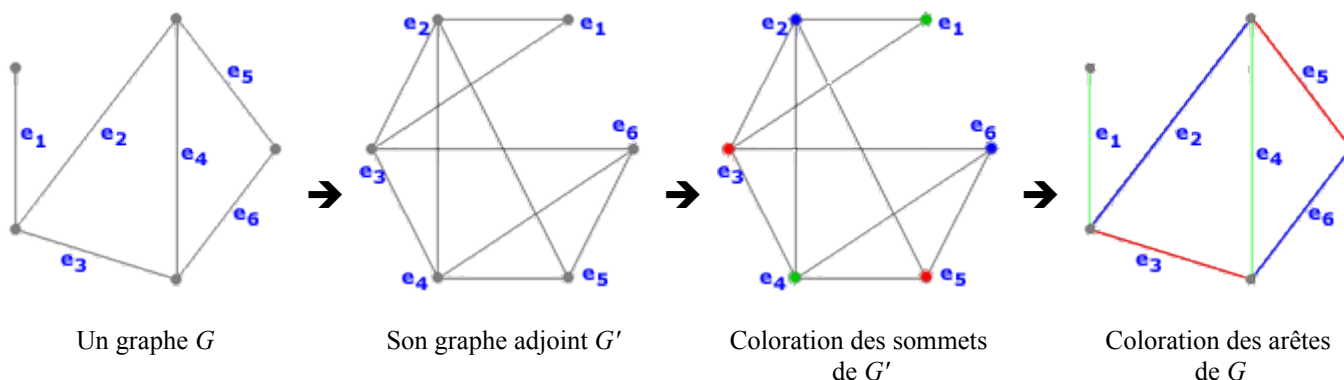
- à chaque arête de  $G = (V, E)$  correspond un sommet de  $G' = (E, F)$
- deux sommets de  $G'$  sont reliés par une arête si les deux arêtes correspondantes de  $G$  sont adjacentes.



Une coloration des arêtes

On peut ensuite appliquer par exemple l'algorithme de Welsh et Powell sur le graphe  $G'$  pour colorer ses sommets. Une fois cela fait, on colorera les arêtes de  $G$  de la même couleur que les sommets correspondants de  $G'$ .

### Exemple de coloration d'arêtes



### Exercice 16.1

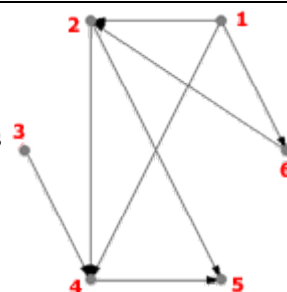
Dans un tournoi d'échecs, chaque engagé doit rencontrer tous les autres. Chaque partie dure une heure. Déterminez la durée minimum du tournoi dans le cas où le nombre d'engagés est 3, 4, 5 ou 6.

## 17. Graphes orientés (digraphes)

En donnant un sens aux arêtes d'un graphe, on obtient un digraphe (ou graphe orienté).

Un digraphe fini  $G = (V, E)$  est défini par l'ensemble fini  $V = \{v_1, v_2, \dots, v_n\}$  ( $|V| = n$ ) dont les éléments sont appelés **sommets**, et par l'ensemble fini  $E = \{e_1, e_2, \dots, e_m\}$  ( $|E| = m$ ) dont les éléments sont appelés arcs.

Un **arc**  $e$  de l'ensemble  $E$  est défini par une paire **ordonnée** de sommets. Lorsque  $e = (u, v)$ , on dira que l'arc  $e$  va de  $u$  à  $v$ . On dit aussi que  $u$  est l'extrémité initiale et  $v$  l'extrémité finale de  $e$ .



### Exercice 17.1

Construire un graphe orienté dont les sommets sont les entiers compris entre 1 et 12 et dont les arcs représentent la relation « être diviseur de ».

---

## 18. Degré d'un sommet d'un digraphe

Soit  $v$  un sommet d'un graphe orienté.

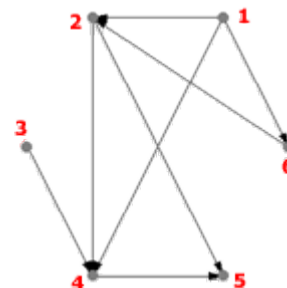
On note  $d^+(v)$  le **degré extérieur** du sommet  $v$ , c'est-à-dire le nombre d'arcs ayant  $v$  comme extrémité initiale.

On note  $d^-(v)$  le **degré intérieur** du sommet  $v$ , c'est-à-dire le nombre d'arcs ayant  $v$  comme extrémité finale.

On a :  $d(v) = d^+(v) + d^-(v)$

### Exercice 18.1

Trouvez les degrés extérieurs et intérieurs de chacun des sommets du graphe ci-contre :



### Exercice 18.2

Soit  $X$  un ensemble de lapins, et  $G$  un graphe orienté ayant  $X$  pour ensemble de sommets. On dit que  $G$  est un « graphe de parenté » si les arcs de  $G$  codent la relation « être l'enfant de... ». Quelles conditions doit nécessairement vérifier  $G$  pour pouvoir être un graphe de parenté ?

---

## 19. Chemins et circuits

Un **chemin** conduisant du sommet  $a$  au sommet  $b$  est une suite de la forme  $(v_0, e_1, v_1, e_2, \dots, v_{k-1}, e_k, v_k)$  où les  $v_i$  sont des sommets ( $v_0=a$  et  $v_k=b$ ) et les  $e_i$  sont des arcs tels que  $e_i$  va de  $v_{i-1}$  à  $v_i$ . Sur le digraphe ci-contre, on peut voir par exemple le chemin  $(v_3, e_3, v_4, e_4, v_5)$ . Par convention, tout chemin comporte au moins un arc.

On appelle **distance** entre deux sommets d'un digraphe la longueur du plus petit chemin les reliant. S'il n'existe pas de chemin entre les sommets  $x$  et  $y$ , on pose  $d(x,y) = \infty$ . Par exemple, sur le digraphe ci-contre,  $d(v_1, v_5) = 2$ ,  $d(v_1, v_6) = 1$ ,  $d(v_6, v_1) = \infty$ .

Un **circuit** est un chemin avec  $u_0 = u_k$ . Le digraphe ci-contre ne contient pas de circuit.

Les notions de longueur, de chemins et de circuits sont analogues à celles des chaînes et des cycles pour le cas non orienté.

### Digraphe fortement connexe

Un digraphe est **fortement connexe**, si toute paire ordonnée  $(a, b)$  de sommets distincts du graphe est reliée par au moins un chemin. En d'autres termes, tout sommet est atteignable depuis tous les autres sommets par au moins un chemin.

On appelle **composante fortement connexe** tout sous-graphe induit maximal fortement connexe (maximal signifie qu'il n'y a pas de sous-graphe induit connexe plus grand contenant les sommets de la composante).

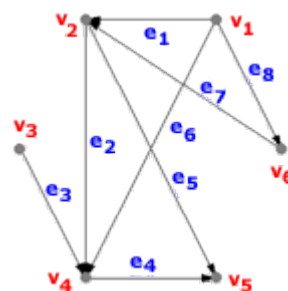
### Exercice 19.1

On souhaite prélever 4 litres de liquide dans un tonneau. Pour cela, nous avons à notre disposition deux récipients (non gradués!), l'un de 5 litres, l'autre de 3 litres.

Comment doit-on procéder ?

### Exercice 19.2 (Jeu de Fan Tan)

Deux joueurs disposent de 2 ou plusieurs tas d'allumettes. À tour de rôle, chaque joueur peut enlever un certain nombre d'allumettes de l'un des tas (selon la règle choisie). Le joueur qui retire la dernière allumette perd la partie.



Modélisez ce jeu à l'aide d'un graphe dans le cas où l'on dispose au départ de deux tas contenant chacun trois allumettes, et où un joueur peut enlever une ou deux allumettes à chaque fois.

Que doit jouer le premier joueur pour gagner la partie à coup sûr ?

### Exercice 19.3

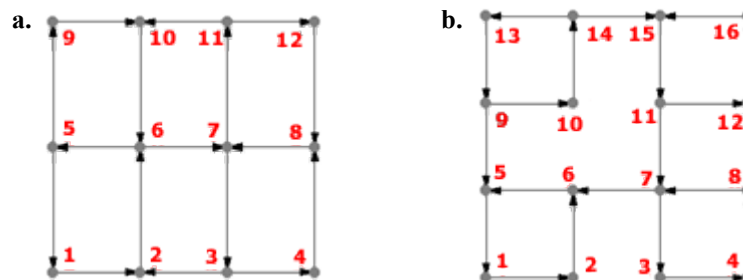
Donnez un algorithme permettant de calculer la distance entre deux sommets  $x$  et  $y$  d'un digraphe connexe.

### Exercice 19.4

Proposez un algorithme qui détermine si un graphe est fortement connexe ou non.

**Indication:** utilisez un système de marquage des sommets.

Les graphes ci-dessous sont-ils fortement connexes ? Si non, donnez leurs composantes fortement connexes.



## 20. Matrice et listes d'adjacences

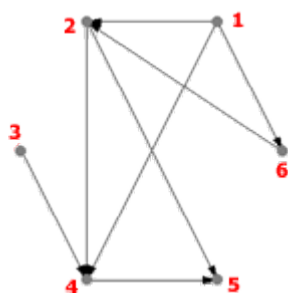
### Matrice d'adjacences

On peut représenter un digraphe par une matrice d'adjacences. Une matrice  $(n \times m)$  est un tableau de  $n$  lignes et  $m$  colonnes.  $(i,j)$  désigne l'intersection de la ligne  $i$  et de la colonne  $j$ .

Dans une matrice d'adjacences, les lignes et les colonnes représentent les sommets du graphe. Un 1 à la position  $(i,j)$  signifie qu'un arc part de  $i$  pour rejoindre  $j$ .

### Exemple

Voici la matrice d'adjacences du digraphe  $G$  :



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

Cette matrice à plusieurs caractéristiques :

1. Elle est carrée : il y a autant de lignes que de colonnes.
2. Il n'y a que des zéros sur la diagonale. Un 1 sur la diagonale indiquerait une boucle.
3. Contrairement au cas non orienté, elle n'est pas symétrique.

Les matrices d'adjacences ont des propriétés intéressantes. On a calculé ci-dessous les matrices  $M^2$  et  $M^3$ . Pour chacune de ces matrices, à quoi correspondent les nombres obtenus ?

$$M^2 = \begin{pmatrix} 0 & 1 & 0 & 1 & 2 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

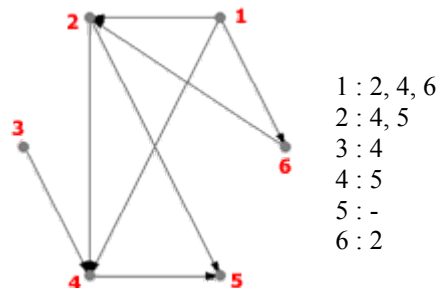
$$M^3 = \begin{pmatrix} 0 & 0 & 0 & 1 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

## Listes d'adjacences

On peut aussi représenter un digraphe en donnant pour chacun de ses sommets la liste des sommets qu'on peut atteindre directement en suivant un arc (dans le sens de la flèche).

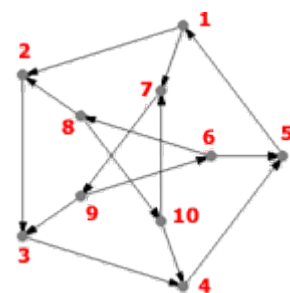
### Exemple

Voici les listes d'adjacences du digraphe  $G$  :



### Exercice 20.1

Décrivez le digraphe  $G$  ci-contre par une matrice d'adjacences et des listes d'adjacences.



## 21. Digraphes sans circuits

### Théorème 9

Le digraphe  $G = (V, E)$  est sans circuits si et seulement si on peut attribuer un nombre  $r(v)$ , appelé le rang de  $v$ , à chaque sommet  $v$  de manière que pour tout arc  $(u, v)$  de  $G$  on ait  $r(u) < r(v)$ .

### Preuve

Si  $G$  comporte un circuit  $C$ , il n'est pas possible de trouver de tels nombres  $r(i)$  car, autrement, considérant  $r(j) = \max \{r(i) \mid i \in C\}$  et l'arc  $(j, k) \in C$  on aurait  $r(j) \leq r(k)$  en contradiction avec la définition de  $r(\cdot)$ .

Réciproquement, si  $G$  n'a pas de circuits, il existe au moins un sommet sans prédécesseurs dans  $G$  (sans cela, en remontant successivement d'un sommet à un prédécesseur, on finirait par fermer un circuit). Ainsi, on peut attribuer séquentiellement des valeurs  $r(\cdot)$  aux sommets du graphe à l'aide de l'algorithme qui suit, ce qui conclura la démonstration.

### Algorithme de calcul du rang

**Donnée :** digraphe  $G = (V, E)$  sans circuit.

**Résultat :** rang  $r(v)$  de chaque sommet  $v \in V$  du digraphe  $G$ .

**Début**

$r := 0, X := V$

$R$  : l'ensemble des sommets de  $X$  sans prédécesseurs dans  $X$

**Tant que**  $X$  n'est pas vide **faire**

**Début**

$r(v) := r$  pour tout sommet  $v \in R$

$X := X - R$

$R$  : l'ensemble des sommets de  $X$  sans prédécesseurs dans  $X$

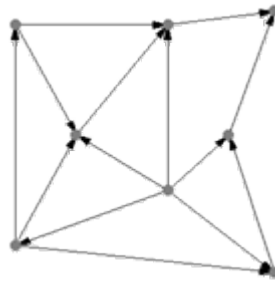
$r := r + 1$

**Fin**

**Fin.**

### Exercice 21.1

Attribuez un rang aux sommets du digraphe ci-dessous en utilisant l'algorithme de calcul du rang :



## 22. Algorithme de Dijkstra

Edgser Wybe Dijkstra (1930-2002) a proposé en 1959 un algorithme qui permet de calculer le **plus court chemin** entre un sommet particulier et tous les autres. Le résultat est une arborescence.

Numérotons les sommets du graphe  $G = (V, E)$  de 1 à  $n$ . Supposons que l'on s'intéresse aux chemins partant du sommet 1. On construit un vecteur  $\lambda = (\lambda(1); \lambda(2); \dots; \lambda(n))$  ayant  $n$  composantes tel que  $\lambda(j)$  soit égal à la longueur du plus court chemin allant de 1 au sommet  $j$ . On initialise ce vecteur à  $c_{1,j}$ , c'est-à-dire à la première ligne de la **matrice des coûts** du graphe, définie comme indiqué ci-dessous :

$$c_{ij} = \begin{cases} 0 & \text{si } i = j \\ \infty & \text{si } i \neq j \text{ et } (i, j) \notin E \\ \delta(i, j) & \text{si } i \neq j \text{ et } (i, j) \in E \end{cases}$$

où  $\delta(i, j)$  est le poids (la longueur) de l'arc  $(i, j)$ . Les  $c_{ij}$  doivent être **strictement positifs**.

On construit un autre vecteur  $p$  pour mémoriser le chemin pour aller du sommet 1 au sommet voulu. La valeur  $p(i)$  donne le sommet qui précède  $i$  dans le chemin.

On considère ensuite deux ensembles de sommets,  $S$  initialisé à  $\{1\}$  et  $T$  initialisé à  $\{2, 3, \dots, n\}$ . À chaque pas de l'algorithme, on ajoute à  $S$  un sommet jusqu'à ce que  $S = V$  de telle sorte que le vecteur  $\lambda$  donne à chaque étape le coût minimal des chemins de 1 aux sommets de  $S$ .



### Description de l'algorithme de Dijkstra

On suppose ici que le sommet de départ (qui sera la racine de l'arborescence) est le sommet 1. Notons qu'on peut toujours renuméroter les sommets pour que ce soit le cas.

**Initialisations**      $\lambda(j) = c_{1,j}$  et  $p(j) = \text{NIL}$ , pour  $1 \leq j \leq n$

**Pour**  $2 \leq j \leq n$  **faire**

**Si**  $c_{1,j} < \infty$  **alors**  $p(j) = 1$ .

$S = \{1\}$  ;  $T = \{2, 3, \dots, n\}$ .

**Itérations**

**Tant que**  $T$  n'est pas vide **faire**

        Choisir  $i$  dans  $T$  tel que  $\lambda(i)$  est minimum

        Retirer  $i$  de  $T$  et l'ajouter à  $S$

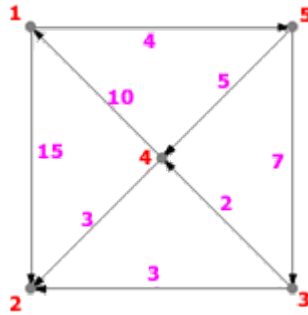
**Pour chaque** successeur  $j$  de  $i$ , avec  $j$  dans  $T$ , **faire**

**Si**  $\lambda(j) > \lambda(i) + \delta(i, j)$  **alors**

$\lambda(j) = \lambda(i) + \delta(i, j)$

$p(j) = i$

## Exemple



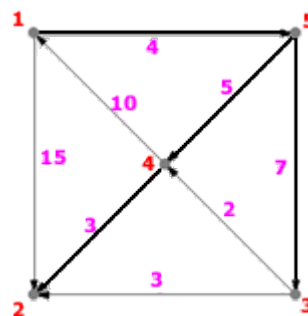
Itérations  $S = \{1\}$  ;  $T = \{2, 3, 4, 5\}$  ;  $\lambda = (0, 15, \infty, \infty, 4)$ ;  $p = (\text{NIL}, 1, \text{NIL}, \text{NIL}, 1)$

1ère itération  $i = 5$  car  $\lambda(5) = \min(15, \infty, \infty, 4) = 4$   
 $S = \{1, 5\}$ ;  $T = \{2, 3, 4\}$   
 les successeurs de 5 dans  $T$  sont 3 et 4  
 $\lambda(3)$  prend la nouvelle valeur  $\min(\infty; \lambda(5) + \delta(5; 3)) = \min(\infty; 4 + 7) = 11$ ;  $p(3) = 5$   
 $\lambda(4)$  prend la nouvelle valeur  $\min(\infty; \lambda(5) + \delta(5; 4)) = 9$ ;  $p(4) = 5$   
 d'où les nouveaux vecteurs  $\lambda = (0, 15, 11, 9, 4)$  et  $p = (\text{NIL}, 1, 5, 5, 1)$

2ème itération  $i = 4$ ;  $\lambda(4) = 9$   
 $S = \{1, 5, 4\}$ ;  $T = \{2, 3\}$   
 le seul successeur de 4 dans  $T$  est 2  
 $\lambda(2)$  prend la nouvelle valeur  $\min(\infty; \lambda(4) + \delta(4; 2)) = \min(15; 9 + 3) = 12$ ;  $p(2) = 4$   
 d'où les nouveaux vecteurs  $\lambda = (0, 12, 11, 9, 4)$  et  $p = (\text{NIL}, 4, 5, 5, 1)$

3ème itération  $i = 3$ ;  $\lambda(3) = 11$   
 $S = \{1, 5, 4, 3\}$ ;  $T = \{2\}$   
 le seul successeur de 3 dans  $T$  est 2  
 $\lambda(2)$  garde sa valeur car  $\min(12; \lambda(3) + \delta(3; 2)) = \min(12; 11 + 3) = 12$   
 d'où les vecteurs inchangés  $\lambda = (0, 12, 11, 9, 4)$  et  $p = (\text{NIL}, 4, 5, 5, 1)$

4ème itération  $i = 2$ ;  $\lambda(2) = 12$   
 $S = \{1, 5, 4, 3, 2\}$ ;  $T = \{\}$ ; FIN  
 $\lambda = (0, 12, 11, 9, 4)$   
 $p = (\text{NIL}, 4, 5, 5, 1)$



Par exemple, le chemin minimal de 1 à 4 est de coût 9.

C'est le chemin 1–5–4, car  $p(4) = 5$  et  $p(5) = 1$ .

## Exercice 22.1

Appliquez l'algorithme de Dijkstra au graphe de l'exemple ci-dessus pour trouver tous les plus courts chemins en partant des sommets 2, 3, 4 et 5.

## Exercice 22.2

Expliquez pourquoi des arcs avec des poids négatifs pourraient poser problème dans la recherche d'un plus court chemin dans un graphe.

## 23. Méthode PERT

Le **problème du plus long chemin** dans les digraphes **sans circuits** trouve une application dans l'ordonnancement et la planification des tâches composant un projet complexe, par exemple la construction d'une maison.

**On fait correspondre à chaque tâche un arc d'un digraphe**, sa durée d'exécution étant égale au poids de cet arc. Le digraphe reflète les précédences requises dans l'exécution du projet. Ainsi, la tâche correspondant à l'arc  $(i, j)$  ne peut commencer que si toutes les tâches correspondant à des arcs  $(k, i)$  ont été complétées. Le digraphe peut contenir des **tâches fictives de durée nulle** afin de forcer certaines précédences.

**Les sommets du digraphe représentent des événements**, début (fin) des activités correspondant aux arcs dont ils sont l'extrémité initiale (finale). Le fait que le digraphe est sans circuit est garant de la faisabilité du projet. En effet, l'existence d'un circuit impliquerait une contradiction dans les précédences : une tâche devant en même temps précéder et succéder une autre!

On supposera dorénavant que les sommets ont déjà été numérotés de 1 à  $n$  de manière compatible avec leurs rangs, c'est-à-dire que  $r(j) > r(i)$  implique  $j > i$  (voir l'algorithme de calcul du rang). En plus, si le digraphe possède plusieurs sommets sans prédécesseurs, on supposera avoir introduit un sommet 1 relié par un arc de durée nulle à chacun de ces sommets. Ce sommet indique le début du projet. De même, si le digraphe possède plusieurs sommets sans successeurs, ceux-ci seront reliés par un arc de durée nulle à un dernier sommet  $n$  (fin du projet). Enfin, on supposera éliminés les arcs parallèles par l'introduction de tâches fictives.

### Algorithme du chemin critique

**Données :** Digraphe  $G = (V, E)$ , sans circuits, des activités avec leur durée  $d_{ik}$ .

**Notations :**

$P(i) = \{k \in V \mid (k, i) \in E\}$  : c'est l'ensemble des sommets prédécesseurs de  $i$ .

$S(i) = \{k \in V \mid (i, k) \in E\}$  : c'est l'ensemble des sommets successeurs de  $i$ .

**Résultat :**

- $\delta_i$  : début au plus tôt des activités correspondant aux arcs  $(i, k)$  partant de  $i$ ,
- $\varphi_i$  : fin au plus tard des activités correspondant aux arcs  $(k, i)$  arrivant à  $i$ ,
- durée du chemin critique.

**Début**

I. Calcul des dates de début au plus tôt (récurrence en avançant dans le projet)

$$\delta_1 := 0$$

**Pour**  $k := 2$  **à**  $n$  **faire**  $\delta_k := \max\{\delta_j + d_{jk} \mid j \in P(k)\}$

II. Calcul des dates de fin au plus tard (récurrence en reculant dans le projet)

$$\varphi_n := \delta_n$$

**Pour**  $k := n-1$  **à**  $1$  **faire**  $\varphi_k := \min\{\varphi_j - d_{kj} \mid j \in S(k)\}$

**Fin.**

### Définitions

Un sommet  $i$  est **critique** si  $\delta_i = \varphi_i$ .

Un arc  $(i, j)$  est **critique** si ses extrémités sont des sommets critiques et  $d_{ij} = \delta_j - \delta_i$ .

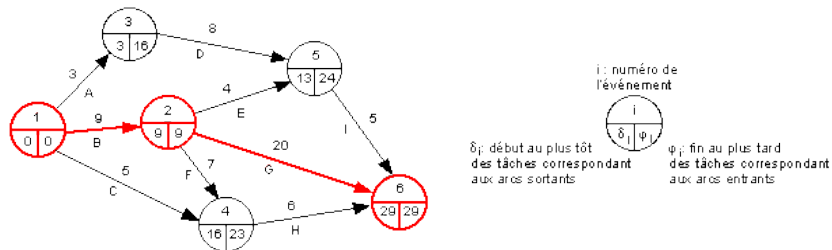
Un **chemin critique** est un chemin de 1 à  $n$  n'utilisant que des arcs critiques, c'est-à-dire des activités telles que tout retard dans leur exécution provoquerait un retard de la fin du projet.

La durée du chemin critique est donnée par  $\delta_n$  (ou par  $\varphi_n$ , les deux valeurs étant toujours égales). Elle correspond à la durée minimale du projet étant données les durées des tâches le composant et les précédences respectives.

### Exemple

Ci-dessous le graphe des précédences obtenu avec l'algorithme du chemin critique. Les sommets et les arcs critiques sont en rouge.

Tâches	Précédences	Durée (jours)
A	-	3
B	-	9
C	-	5
D	A	8
E	B	4
F	B	7
G	B	20
H	C, F	6
I	D,E	5



### Exercice 23.1

Refaites le graphe des précérences de l'exemple en utilisant l'algorithme du chemin critique.

### Exercice 23.2

La rénovation du séjour d'un appartement se décompose en plusieurs tâches décrites dans le tableau ci-dessous. Ce dernier donne également les précérences à respecter lors de la planification des travaux ainsi qu'une estimation de la durée de chacune des tâches.

Tâches	Précérences	Durée (jours)
A Enlèvement des portes	-	1/2
B Ponçage et peinture des portes	A	3
C Pose des portes	B, J	1/2
D Arrachage des papiers peints	-	1
E Tirage des fils électriques	D	1
F Pose des prises	E, H, I	1/2
G Ragréage des murs	E, A	2
H Peinture du plafond	G	2
I Pose des papiers peints	G	3
J Peinture des cadres	H, I	1
K Arrachage de la moquette	H, I, J	1/2
L Ponçage du parquet	K	1
M Imprégnation et séchage du parquet	L, F	4
N Peinture du balcon	-	2
O Changement des protections solaires	N	1

1. Représentez le graphe des précérences de ces travaux de rénovation.
2. Déterminez une durée totale minimale de rénovation en exhibant un chemin critique dans le graphe précédent.



## 24. Introduction aux chaînes de Markov

Généralement, un **processus stochastique** est une suite d'expériences dont le résultat dépend du hasard. Ici, nous admettrons qu'en certains temps  $0, 1, 2, \dots, t$ , nous observons un système. Celui-ci peut se trouver dans l'un des états d'une collection finie d'états possibles. L'observation du système est ainsi considérée comme une expérience dont le résultat (aléatoire) est l'état dans lequel se trouve le système.

Nous supposons que nous connaissons pour chaque paire d'états  $i$  et  $j$ , et pour chaque instant  $t$ , la probabilité  $p_{ij}(t)$  que le processus soit dans l'état  $j$  à l'instant  $t+1$  étant donné qu'il se trouve dans l'état  $i$  à l'instant  $t$ . De plus, la probabilité  $p_{ij}(t)$  sera supposée **ne pas dépendre de  $t$** .

Un tel processus est appelé **chaîne de Markov** (à temps discret et avec un ensemble fini d'états), du nom de son inventeur *Andrei Andrejevich Markov* (1856-1922), dont vous voyez le visage ci-contre.



Avec ces hypothèses, nous pouvons décrire le système en donnant l'ensemble  $\{u_1, \dots, u_m\}$  des états  $u_i$  possibles et une matrice  $P$  de dimensions  $m \times m$  dont le terme  $p_{ij}$  est la probabilité que le processus soit dans l'état  $j$  à l'instant  $t+1$  étant donné qu'il se trouve dans l'état  $i$  à l'instant  $t$ , **pour tout  $t$** .  $P$  est appelée **matrice de transition** du système.

On représente généralement  $P$  par un graphe orienté  $G$  dont les sommets correspondent aux  $m$  états et les arcs aux couples ordonnés d'états  $(i, j)$  tels que  $p_{ij} > 0$ .

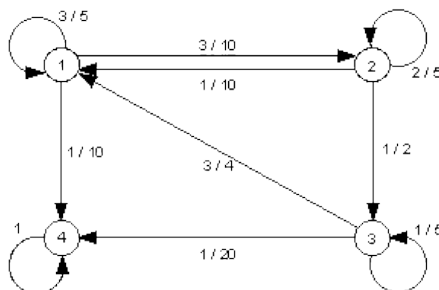
### Exemple

Pour représenter le passage d'une molécule de phosphore dans un écosystème, nous considérerons quatre états possibles :

1. la molécule est dans le sol,
2. la molécule est dans l'herbe,
3. la molécule a été absorbée par du bétail,
4. la molécule est sortie de l'écosystème.

La matrice de transition est la suivante : 
$$P = \begin{pmatrix} \frac{3}{5} & \frac{3}{10} & 0 & \frac{1}{10} \\ \frac{1}{10} & \frac{2}{5} & \frac{1}{2} & 0 \\ \frac{3}{4} & 0 & \frac{1}{5} & \frac{1}{20} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Remarquez que **la somme de chaque ligne vaut 1**. Cette matrice correspond au graphe ci-dessous :



### Propriété 1

La probabilité  $p_{ij}^{(t)}$  que le système soit dans l'état  $j$  au temps  $t$  sachant qu'il était dans l'état  $i$  au temps 0 est donné par  $(P^t)_{i,j}$  (le terme  $i,j$  de la  $t$ -ième puissance de  $P$ ).

Si on ne connaît pas l'état initial, on peut donner un vecteur de probabilité  $\pi^{(0)} = (\pi_1^{(0)}, \dots, \pi_m^{(0)})$  où  $\pi_i^{(0)}$  est la probabilité que le système se trouve dans l'état  $i$  au temps 0. Si  $\pi^{(t)}$  est le vecteur donnant les probabilités d'occupation des états au temps  $t$  (autrement dit la distribution), nous avons :

## Propriété 2

$$\pi^{(t)} = \pi^{(0)} P^t$$

### Exercice 24.1

Écrivez dans *Mathematica* un programme qui simule une chaîne de Markov donnée par sa matrice de transition. Comme résultat, on veut la liste des états successifs du processus et le pourcentage de fois où le processus a été dans chacun des états.

### Exercice 24.2

Utilisez votre programme de l'exercice 24.1 pour simuler la chaîne de Markov donnée en exemple.

Partant du vecteur initial  $\pi^{(0)} = (1, 0, 0, 0)$ , déterminez  $\pi^{(10)}$ ,  $\pi^{(100)}$  et  $\pi^{(1000)}$  par calcul, puis par simulation en utilisant votre programme. Vous calculerez  $P^t$  avec *Mathematica*.

### Exercice 24.3

Un individu vit dans un milieu où il est susceptible d'attraper une maladie par piqûre d'insecte. Il peut être dans l'un des trois états suivants : immunisé (I), malade (M), non malade et non immunisé (S). D'un mois à l'autre, son état peut changer selon les règles suivantes :

- étant immunisé, il peut le rester avec une probabilité 0.9 ou passer à l'état S avec une probabilité 0.1 ;
- étant dans l'état S, il peut le rester avec une probabilité 0.5 ou passer à l'état M avec une probabilité 0.5 ;
- étant malade, il peut le rester avec une probabilité 0.2 ou passer à l'état I avec une probabilité 0.8.

Tracez un graphe probabiliste pour décrire cette situation et écrivez la matrice de transition. Calculez l'état de probabilité de l'individu au bout de trois mois, de six mois, d'un an, de deux ans, pour chacune des situations suivantes :

- au départ, il est immunisé (I);
- au départ, il est non malade et non immunisé (S);
- au départ, il est malade (M).

Pouvez-vous donner des éléments sur la proportion d'individus malades dans la population étudiée ?

## 25. Distribution limite

On constate souvent (par exemple à l'exercice 24.2) que la distribution  $\pi(t)$  converge vers une distribution limite  $\pi$  si  $t \rightarrow \infty$ . Si tel est le cas, on dit que cette dernière définit un **régime permanent** du processus stochastique. Le régime permanent n'est pas influencé par le choix de la distribution initiale.

### Existence d'une distribution limite

#### Propriété 3

Si la matrice de transition  $P$  est telle qu'une au moins de ses puissances n'a que des termes strictement positifs, alors  $\pi(t) \rightarrow \pi$  quelle que soit la distribution initiale  $\pi(0)$  et  $P^t \rightarrow P^*$  lorsque  $t \rightarrow \infty$ .  $\pi$  est un vecteur de probabilité strictement positif et  $P^*$  une matrice dont toutes les lignes sont identiques au vecteur limite  $\pi$ . En plus,  $\pi P^* = \pi$ .

La démonstration de cette condition d'existence dépasse le cadre de ce cours.

### Exercice 25.1

Soit la matrice stochastique  $P = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 1 & 0 \end{pmatrix}$

Montrez que la chaîne de Markov définie par  $P$  converge et calculez la distribution limite.

### Exercice 25.2

Soit la matrice stochastique  $P = \begin{pmatrix} \frac{3}{5} & \frac{3}{10} & 0 & \frac{1}{10} \\ \frac{1}{10} & \frac{2}{5} & \frac{1}{2} & 0 \\ \frac{3}{4} & 0 & \frac{1}{5} & \frac{1}{20} \\ 0 & 0 & 0 & 1 \end{pmatrix}$ .

Montrez que la chaîne de Markov définie par  $P$  converge et calculez la distribution limite.

### Exercice 25.3

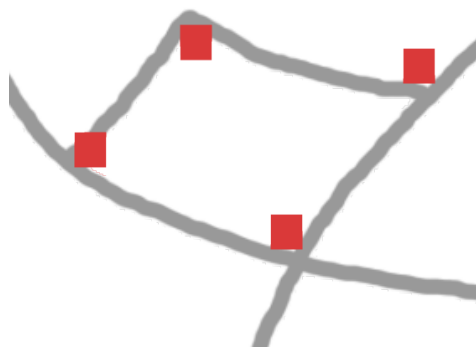
Un service de météo a constaté après de longues années que le temps qu'il fera demain dépend essentiellement du temps qu'il faisait hier et du temps qu'il fait aujourd'hui. Les probabilités de transition ont été établies ainsi :

Hier	Aujourd'hui	Demain	
		Beau	Mauvais
Beau	Beau	0.8	0.2
Beau	Mauvais	0.4	0.6
Mauvais	Beau	0.6	0.4
Mauvais	Mauvais	0.1	0.9

1. Modélisez ce processus à l'aide d'une chaîne de Markov.
2. Calculez le nombre moyen de jours de beau temps par année.

### Exercice 25.4

Un ivrogne se déplace dans les quatre bistrot du village (voir plan ci-dessous) d'une manière bien personnelle : en sortant d'un bistrot, il lance une pièce de monnaie pour savoir dans lequel des deux autres bistrot les plus proches il entrera.



1. Modélisez ce processus à l'aide d'une chaîne de Markov.
2. Montrez que cette chaîne de Markov n'a pas de distribution limite.

## 26. Chaîne absorbante

Un **état absorbant** est un état que l'on ne quitte plus lorsqu'on y pénètre. Autrement dit, l'état  $j$  est absorbant si  $p_{jj} = 1$ .

Une **chaîne de Markov** est **absorbante** si et seulement si :

1. il y a au moins un état absorbant,
2. de tout état non absorbant, on peut atteindre un état absorbant.

Par exemple, l'état 4 du premier exemple de ce chapitre est un état absorbant. Comme on peut atteindre cet état depuis tous les autres, la chaîne de Markov est absorbante.

### Propriété 4

Pour toute chaîne de Markov absorbante et pour tout état de départ, la probabilité de se trouver dans un état absorbant au temps  $t$  tend vers 1 lorsque  $t$  tend vers l'infini.

### Délais d'absorption et probabilité d'absorption

Lorsque l'on a affaire à une chaîne de Markov absorbante, on est généralement intéressé par les deux questions suivantes :

- Combien de temps faudra-t-il en moyenne pour arriver dans un état absorbant, étant donné son état initial ?
- S'il existe plusieurs états absorbants, quelle est la probabilité de tomber dans un état absorbant donné ?

### Forme canonique de $P$

Si une chaîne de Markov est absorbante, on placera au début les états absorbants ; on aura alors une matrice de transition de la forme suivante ( $I$  est une matrice unité et  $0$  une matrice de 0) :

$$\begin{array}{cc} & \begin{array}{cc} \text{absorbants} & \text{non absorbants} \end{array} \\ \begin{array}{c} \text{absorbants} \\ \text{non absorbants} \end{array} & \left( \begin{array}{c|c} I & 0 \\ \hline R & Q \end{array} \right) \end{array}$$

Dans l'exemple du phosphore vu précédemment, nous avons (les numéros des états sont en rouge) :

$$\begin{array}{c} \text{4} \quad \text{1} \quad \text{2} \quad \text{3} \\ \begin{array}{c} \text{4} \\ \text{1} \\ \text{2} \\ \text{3} \end{array} \left( \begin{array}{c|ccc} 1 & 0 & 0 & 0 \\ \hline \frac{1}{10} & \frac{3}{5} & \frac{3}{10} & 0 \\ 0 & \frac{1}{10} & \frac{2}{5} & \frac{1}{2} \\ \frac{1}{20} & \frac{3}{4} & 0 & \frac{1}{5} \end{array} \right) \end{array}$$

La matrice  $N = (I - Q)^{-1}$  est appelée la **matrice fondamentale** de la chaîne absorbante.

### Propriété 5

Le nombre moyen  $e_{ij}$  de passages à l'état  $j$  (non absorbant) avant l'absorption quand on part de l'état  $i$  (non absorbant) est donnée par  $e_{ij} = (N)_{ij}$ .

Le nombre moyen d'étapes avant absorption sachant que l'on part de l'état  $i$  (non absorbant) est la somme des termes de la  $i$ -ème ligne de  $N$ .

Toujours dans l'exemple du phosphore, on a :

$$Q = \begin{pmatrix} \frac{3}{5} & \frac{3}{10} & 0 \\ \frac{1}{10} & \frac{2}{5} & \frac{1}{2} \\ \frac{3}{4} & 0 & \frac{1}{5} \end{pmatrix}, \quad I - Q = \begin{pmatrix} \frac{2}{5} & -\frac{3}{10} & 0 \\ -\frac{1}{10} & \frac{3}{5} & -\frac{1}{2} \\ -\frac{3}{4} & 0 & \frac{4}{5} \end{pmatrix}, \quad \text{d'où } N = (I - Q)^{-1} = \begin{pmatrix} \frac{320}{37} & \frac{160}{37} & \frac{100}{37} \\ \frac{910}{111} & \frac{640}{111} & \frac{400}{111} \\ \frac{300}{37} & \frac{150}{37} & \frac{140}{37} \end{pmatrix}$$

D'où le nombre moyen d'étapes avant absorption en partant de l'état 1:  $(320 + 160 + 100) / 37 = 15.67$

## Propriété 6

Dans une chaîne de Markov absorbante avec  $P$  mise sous forme canonique, le terme  $b_{ij}$  de la matrice  $B = N \cdot R$  est la probabilité d'absorption par l'état absorbant  $j$  sachant que l'on part de l'état  $i$ .

Dans l'exemple du phosphore, on a :

$$R = \begin{pmatrix} \frac{1}{10} \\ 0 \\ \frac{1}{20} \end{pmatrix} \quad \text{d'où } B = N \cdot R = \begin{pmatrix} \frac{320}{37} & \frac{160}{37} & \frac{100}{37} \\ \frac{910}{111} & \frac{640}{111} & \frac{400}{111} \\ \frac{300}{37} & \frac{150}{37} & \frac{140}{37} \end{pmatrix} \cdot \begin{pmatrix} \frac{1}{10} \\ 0 \\ \frac{1}{20} \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

La probabilité d'être absorbé par l'unique état absorbant est 1 quel que soit l'état initial !

### Exercice 26.1

Considérons un joueur qui possède 2 francs initialement. À chaque étape du jeu il peut gagner 1 franc avec probabilité  $p$  ou perdre 1 franc avec probabilité  $1-p$ . Il s'arrête lorsqu'il a gagné 4 francs ou lorsqu'il a tout perdu.

1. Représentez cette expérience par une chaîne de Markov.
2. Avec  $p = 1/3$ , calculez la probabilité de la ruine du joueur.
3. Avec  $p = 1/3$ , calculez la longueur moyenne d'une partie.

### Exercice 26.2

Monsieur X se rend au Salon du Livre de Rigoleville dans l'espoir de trouver enfin un exemplaire du livre de Stendhal *Le Rose et le Vert*. Le Salon compte cinq stands et les organisateurs se sont amusés aux cours des années précédentes à construire la matrice des probabilités de transition des visiteurs d'un stand à un autre :

de \ à	Stand 1	Stand 2	Stand 3	Stand 4	Stand 5
Stand 1	0	0.8	0	0	0.2
Stand 2	0.2	0	0.5	0.3	0
Stand 3	0	0	0	0.6	0.4
Stand 4	0.9	0	0.1	0	0
Stand 5	0.8	0	0	0.2	0

Sachant que seuls les stands 4 et 5 disposent du livre recherché et que Monsieur X commence par visiter le stand 1, quelle est la probabilité qu'il achète son livre au stand 4 plutôt qu'au stand 5 (Monsieur X achètera le premier exemplaire qu'il trouvera) ?

### Exercice 26.3

Considérons une série de jets d'une pièce de monnaie normale. On notera P pour pile et F pour face.

1. Quelle est la probabilité d'obtenir une séquence PFP avant une séquence PPP ?
2. Quel est le nombre de jets nécessaires en moyenne pour réaliser l'une des deux séquences PFP et PPP ?

## Exercice 26.4 : la parade nuptiale des bourdons

Cet exercice est tiré du livre donné en référence. On s'y reportera pour avoir tous les détails de l'expérience. On ne présente ici que le minimum nécessaire à l'exercice.

### Observation d'une séance d'accouplement

Une séance d'accouplement peut se décomposer en 7 phases :

- **Départ (D)** : mise en contact des bourdons mâle et des reines.
- **Approche (App)** : un mâle se dirige vers la reine. Il s'approche à courte distance. Il est le comportement le plus fréquent et souvent suivi d'une récursive.
- **Inspection de la femelle (IF)** : le mâle suit la reine avec ses antennes tendues vers elle. Il inspecte souvent la reine au niveau de la tête (région où se trouvent les glandes produisant les phéromones sexuelles), mais parfois au niveau de l'abdomen.
- **Tentative d'accouplement (T)** : le mâle s'approche de la reine, il s'accroche à elle. Il frotte de ses pattes antérieures l'extrémité de l'abdomen de la femelle. Il sort ses génitalia (appareil reproducteur) et tente de pénétrer la reine.
- **Accouplement (Acc)** : lors de l'accouplement, le comportement du mâle se caractérise par des mouvements de battements des pattes sur l'extrémité de l'abdomen de la reine.
- **Sortie par abandon du mâle (SA)** : lors de la séquence de 15 minutes, le bourdon mâle peut adopter un comportement indifférent vis-à-vis de la reine; il sort de la parade nuptiale et n'y revient jamais.
- **Sortie pour dépassement du temps (ST)** : l'observation est limitée à 15 minutes. Après cette durée, la probabilité d'accouplement peut être considérée comme presque nulle.



Approche du mâle



Inspection de la reine par le mâle



Tentative d'accouplement



Accouplement

### Résultats des observations

Voici les statistiques obtenues pour 78 séances d'accouplement en laboratoire.

	suivi de						
	App	T	IF	Acc	ST	SA	Total
D	78	0	0	0	0	0	78
App	614	202	87	0	16	8	927
IF	83	0	0	0	3	1	87
T	152	0	0	35	7	8	202

### Travail

1. Dessinez le graphe de transitions d'une parade nuptiale de bourdons.
2. Calculez les probabilités de transition d'un état à un autre et ajoutez-les au graphe.
3. Donnez la matrice correspondante de la chaîne de Markov.
4. Adaptez votre programme simulant une chaîne de Markov (voir exercice 24.1) à la situation présente. Utilisez ce programme pour simuler une parade nuptiale de bourdons.
5. Cette chaîne de Markov est une chaîne absorbante. Quel est le nombre moyen d'étapes avant absorption ? Trouvez le résultat théoriquement et par simulation.

Référence: **Mathématique & biologie. Une expérience pluridisciplinaire**, Éditions De Boeck, Bruxelles, 2003, chapitre 7

---

# Lexique sur les graphes

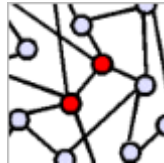
Entre parenthèses, le terme anglais.

## Acyclique (Acyclic)

Un [graphe](#) est acyclique s'il ne contient aucun [cycle](#).

## Adjacent (Adjacent)

Deux [sommets](#) sont adjacents s'ils sont reliés par une [arête](#). On qualifie souvent de **voisins** deux sommets adjacents. Voici deux sommets adjacents :



## Arborescence (Rooted tree)

[Arbre](#) avec un [sommets](#) distingué  $r$  (la [racine](#)).

## Arbre (Tree)

[Graphe connexe](#) ne contenant aucun [cycle](#).

## Arbre couvrant (Spanning tree)

Un [sous-graphe maximum](#) d'un [graphe](#) qui est aussi un [arbre](#). On parle aussi d'arbre de recouvrement.

## Arbre de recherche dichotomique (Binary search tree)

[Arbre binaire](#) qui a été étiqueté avec des nombres de sorte que le [fils](#) de droite d'un sommet  $s$  et tous ses descendants aient des numéros plus petits que le numéro de  $s$ , et le fils de gauche de  $s$  et tous ses descendants ont des numéros plus grands que celui de  $s$ .

## Arbre n-aire (N-ary tree)

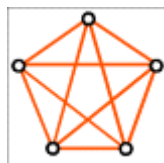
Un [arbre](#) où chaque [sommets](#) a 0 ou  $n$  [fils](#). Quand  $n = 2$ , on parle d'**arbre binaire**.

## Arc (Arc)

Une [arête](#) orientée d'un [digraphe](#).

## Arête (Edge)

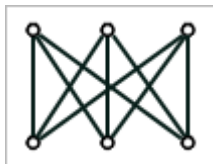
Une arête relie deux [sommets](#) dans un [graphe](#). Nous appelons ces deux sommets les **extrémités** de l'arête. Voici les arêtes d'un graphe (en rouge) :



## Biparti (Bipartite)

Un [graphe](#) est biparti si ses [sommets](#) peuvent être divisés en deux ensembles  $X$  et  $Y$ , de sorte que toutes les [arêtes](#) du graphe relient un sommet dans  $X$  à un sommet dans  $Y$ . Les [arbres](#) sont des exemples des graphes bipartis. Si  $G$  est biparti, il est habituellement noté par  $G = (X, Y, E)$ , où  $E$  est l'ensemble des arêtes.

$K_{3,3}$  est un graphe biparti :

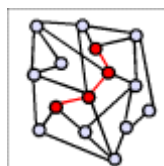


### Boucle (Loop)

Arête ou arc partant d'un sommet et allant vers lui-même. Les boucles ne sont pas autorisées dans les graphes et digraphes simples.

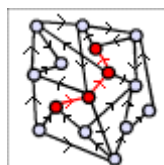
### Chaîne (Chain)

Une chaîne dans un graphe est une suite de sommets reliés par des arêtes. La **longueur** d'une chaîne est le nombre d'arêtes utilisées, ou, ce qui revient au même, le nombre de sommets utilisés moins un. Une chaîne **simple** ne peut pas visiter le même sommet deux fois. Voici un exemple d'une chaîne simple :



### Chemin (Path)

Un chemin dans un digraphe est une suite de sommets reliés les uns aux autres par des arcs. La **longueur** du chemin est le nombre d'arcs utilisés, ou le nombre de sommets moins un. Un chemin **simple** ne peut pas visiter le même sommet plus d'une fois. Un chemin **fermé** a pour dernier sommet le premier. Voici un exemple de chemin :



### Circuit (Circuit)

Dans un digraphe, un circuit est un chemin fermé simple.

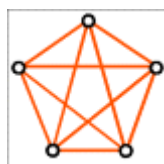
### Clique (Clique)

Sous-graphe complet d'un graphe  $G$ . L'ordre de la plus grande clique de  $G$  est noté  $\omega(G)$ . Prononcer « oméga de  $G$  ».

### Complet (Complete)

Dans un graphe complet, toutes les paires de sommets sont adjacentes. Un graphe complet à  $n$  sommets est noté  $K_n$  (le  $K$  est en l'honneur de Kuratowski, un pionnier de la théorie des graphes).

Voici le graphe complet sur cinq sommets, noté  $K_5$  :



### Composante connexe (Connected component)

Dans un graphe, une composante connexe est un sous-graphe induit maximal connexe. Maximal signifie qu'il n'y a pas de sous-graphe induit connexe plus grand contenant les sommets de la composante.



## Condensé (Condensed)

Étant donné un [graphe](#)  $G$ , si deux [sommets](#) de  $G$  sont fusionnés et si tous les [boucles](#) ou arêtes [multiples](#) créés par cette fusion sont enlevées, le graphe résultant s'appelle **le graphe condensé**.

## Connexe (Connected)

Un [graphe](#) connexe est un graphe dans lequel chaque paire de [sommets](#) est reliée par une [chaîne](#). Un graphe qui n'est pas connexe est dit **non connexe**, et se décompose en [composantes connexes](#).

## Corde (Chord)

On appelle corde d'un [cycle](#) élémentaire une [arête](#) qui relie deux [sommets](#) non consécutifs de ce cycle.

## Couplage ou appariement (Matching)

Un couplage est un ensemble d'[arêtes](#) tel que chaque [sommets](#) du [graphe](#) appartient à au plus une arête de cet ensemble.

## Couplage parfait (Perfect matching)

Dans un [graphe](#) à  $2n$  [sommets](#), un [couplage](#) avec  $n$  [arêtes](#) est dit parfait. Chaque sommet du graphe est [saturé](#) par un couplage parfait.

## Cycle (Cycle)

Dans un [graphe](#), un cycle est une [chaîne](#) simple dont les extrémités coïncident. On ne rencontre pas deux fois le même [sommets](#), sauf celui choisi comme sommet de départ et d'arrivée.

## Degré (Degree)

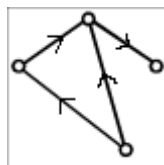
Le degré d'un [sommets](#) est la taille de son [voisinage](#). Le degré d'un [graphe](#) est le degré maximum de tous ses sommets.

## Diamètre (Diameter)

Le diamètre d'un [graphe](#) est la plus longue des [distances](#) entre deux [sommets](#) de ce graphe.

## Digraphe (Digraph)

Un digraphe est un [graphe](#) dans lequel les arêtes sont orientées et appelées [arcs](#). Plus formellement, un digraphe est un ensemble de [sommets](#) ainsi qu'un ensemble de paires ordonnées des sommets, appelées les arcs. Voici un digraphe sur 4 sommets :



## Distance (Distance)

La distance entre deux [sommets](#) est la longueur de la plus courte [chaîne](#) entre eux.

## Étiquette (Label)

Les étiquettes sont simplement des noms donnés aux [sommets](#) et aux [arêtes](#) de façon à pouvoir les différencier. L'étiquetage des sommets est arbitraire.

## Eulérien (Eulerian)

Une [chaîne](#) ou un [circuit](#) est dit **eulérien** si chaque arête du graphe y apparaît exactement une fois.

## Fermeture (Closure)

La fermeture d'un [graphe](#)  $G$  à  $n$  [sommets](#) est le graphe obtenu à partir de  $G$  en ajoutant progressivement des [arêtes](#) entre les sommets non [adjacents](#) dont la somme des [degrés](#) est au moins égale à  $n$ , jusqu'à ce que ceci ne puisse plus être fait. Plusieurs résultats sur l'existence des [circuits hamiltoniens](#) se rapportent à la fermeture d'un graphe.

## Feuille (Leaf)

[Sommets](#) de [degré](#) 1. Aussi appelé [sommets pendants](#).

## Fils (Offspring)

Dans un [arbre](#), les sommets [adjacents](#) à un [sommets](#) donné du [niveau](#) supérieur sont appelés **fils** de ce sommet. Les **descendants** d'un sommet sont les sommets qui sont le fils, ou le fils du fils, etc., d'un sommet donné.

## Forêt (Forest)

[Graphe](#) qui ne contient aucun [cycle](#). Les [composantes connexes](#) d'une forêt sont des [arbres](#).

## Fortement connexe (Strongly Connected)

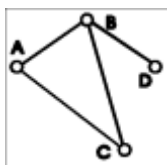
Dans un [digraphe](#) **fortement connexe**, chaque [sommets](#) peut être atteint depuis n'importe quel autre par un [chemin](#).

## Graphe (Graph)

Un graphe est un ensemble de points, dont certaines paires sont reliées par des lignes. Les points sont appelés sommets et les lignes sont nommées arêtes.

Plus formellement, un graphe est composé de deux ensembles, l'ensemble des [arêtes](#) ( $E$ ) et l'ensemble des [sommets](#) ( $V$ ). L'ensemble des sommets est simplement une collection d'[étiquettes](#) qui permettent de distinguer un sommet d'un autre. L'ensemble des arêtes est constitué de paires non ordonnées d'étiquettes de sommets.

Voici la représentation graphique d'un graphe et les ensembles définissant le graphe :



$V = \{A, B, C, D\}$   
- l'ensemble des sommets

$E = \{(A,B), (A,C), (B,C), (B,D)\}$   
- l'ensemble des arêtes

Une représentation graphique

## Hamiltonien (Hamiltonian)

Une [chaîne](#) ou un [cycle](#) est dit **hamiltonien** si chaque [sommets](#) du [graphe](#) y apparaît exactement une fois. Les [chemins](#) et les [circuits](#) des [digraphes](#) sont dits hamiltoniens sous les mêmes conditions. Un graphe contenant un circuit hamiltonien ou un digraphe contenant un cycle hamiltonien est appelé **graphe ou digraphe hamiltonien**.

## Hauteur (Height)

La hauteur d'un [arbre](#) est la longueur de la plus longue [chaîne simple](#) partant de la [racine](#) de l'arbre.

## Homéomorphe (Homeomorphic)

Deux [graphes](#) sont homéomorphes s'ils peuvent tous les deux être obtenus à partir d'un graphe commun en remplaçant les [arêtes](#) par des [chaînes simples](#).

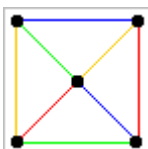
## Incident (Incident)

Un [sommet](#) est incident à une [arête](#) s'il est situé à une des deux extrémités de cette arête. Inversement, une [arête](#) est incidente à un [sommet](#) si elle « touche » ce sommet.

## Indice chromatique (Chromatic index ?)

L'indice chromatique d'un [graphe](#) est le plus petit nombre  $k$  pour lequel il existe une  $k$ -coloration des [arêtes](#). L'indice chromatique du graphe  $G$  est noté par  $\chi(G)$  [ $\chi$  est la lettre grecque khi].

Dans l'exemple ci-dessous, l'indice chromatique vaut 4.



## Isomorphe (Isomorphic)

Deux [graphes](#) sont isomorphes si ce sont les mêmes graphes, dessinés différemment. Deux graphes sont isomorphes si on peut étiqueter les deux graphes avec les mêmes [étiquettes](#) de sorte que chaque sommet ait exactement les mêmes [voisins](#) dans les deux graphes. Voici deux graphes isomorphes :



## Isthme (Bridge)

[Arête](#) dont la suppression augmente le nombre de [composantes connexes](#) du graphe.

## $k$ -colorable ( $k$ -colorable)

Un [graphe](#) est dit  $k$ -colorable si à chacun de ses [sommets](#) peut être assignée une parmi  $k$  couleurs de sorte qu'à deux sommets [adjacents](#) soit assignée une couleur différente. Cette assignation est appelée **coloration**.

## Liste d'adjacences (Adjacency Structure)

Une représentation d'un graphe ou d'un digraphe qui énumère, pour chaque sommet, tous les sommets qui sont [adjacents](#) au sommet donné.

## Liste d'arcs (Arc List)

Une représentation d'un [digraphe](#) utilisant les [arcs](#) du digraphe. Ce peut être une liste de paires ordonnées de sommets, ou deux listes triées avec le [sommet](#) de départ dans une liste et le sommet de fin à la position correspondante de la deuxième liste.

## Matrice d'adjacences (Adjacency Matrix)

Une matrice carrée contenant des 0 et des 1, dont les lignes et les colonnes sont classées par [sommets](#). Un 1 en position  $(i, j)$  signifie qu'il y a une [arête](#) (ou [arc](#)) du sommet  $i$  au sommet  $j$ . Un 0 indique qu'il n'y a aucune arête ou arc. Une matrice d'adjacences peut être utilisée pour des [graphes](#) et des [digraphes](#).

## Matrice d'incidences (Incidence Matrix)

Une matrice contenant des 0 et des 1 dont les lignes sont indexées par les [sommets](#) du [graphe](#) et dont les colonnes sont indexées par les [arêtes](#). Un 1 à la position  $(i, j)$  de la matrice signifie que le sommet  $i$  est une extrémité de l'arête  $j$ . Un 0 indique que ce n'est pas le cas.

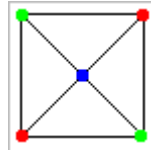
## Noeud (Node)

Autre mot pour [sommet](#).

## Nombre chromatique (Chromatic number)

Le nombre chromatique d'un [graphe](#) est le plus petit nombre  $k$  pour lequel il existe une [k-coloration des sommets](#). Le nombre chromatique du graphe  $G$  est noté par  $\gamma(G)$  [ $\gamma$  est la lettre grecque gamma].

Dans l'exemple ci-dessous, le nombre chromatique vaut 3.



## Ordre (Order)

L'ordre d'un [graphe](#) est le nombre de ses [sommets](#).

## Ordre topologique (Topological Order)

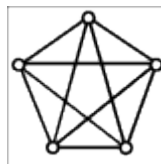
Un ordre topologique d'un [digraphe](#) est un [étiquetage](#) des [sommets](#) avec des entiers consécutifs de sorte que chaque [arc](#) soit orienté d'un numéro plus petit vers un plus grand.

## Orientation (Orientation)

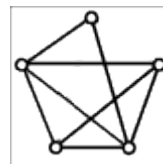
Une assignation de direction aux [arêtes](#) d'un [graphe](#). Une arête orientée est un [arc](#). Le graphe auquel on a donné une orientation est dit **graphe orienté** et est un [digraphe](#).

## Partiel (Spanning Subgraph)

Le [graphe](#) obtenu en enlevant des [arêtes](#) d'un graphe  $G$  est appelé graphe partiel :



Un graphe



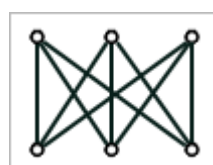
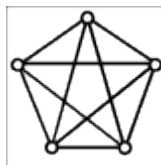
Un graphe partiel

## Pendant (Pendant)

Un [sommets](#) est pendant s'il est de [degré](#) 1. Aussi appelé [feuille](#) si le graphe est un [arbre](#).

## Planaire (Planar)

Un [graphe](#) planaire est un graphe que l'on peut dessiner sur une surface plate sans que ses [arêtes](#) se croisent. Les graphes que l'on ne peut pas dessiner sans croisements sont dits non planaires. Tout graphe contenant l'un ou l'autre des deux sous-graphes ci-dessous est non planaire :



## Racine (Root)

[Sommets](#) distingué d'un [arbre](#). En distinguant un sommet d'un arbre, on obtient une [arborescence](#).

## Rang (Level)

Dans une [arborescence](#), les [sommets](#) à la même [distance](#) de la [racine](#) sont dits être au même **rang**. La racine est par convention au rang 0 et la [hauteur](#) de l'arbre est le rang maximum.

## Réduit (Reduced)

Si une [arête](#)  $a$ , est enlevée d'un [graphe](#)  $G$ , le graphe résultant, noté  $G'_a$ , est appelé le **graphe réduit**.

## Régulier (Regular)

Dans un [graphe](#) régulier, tous les [sommets](#) ont le même [degré](#). Si le degré commun est  $k$ , alors on dit que le graphe est  **$k$ -régulier**.

## Saturé (Saturated)

Un [sommets](#) appartenant à une [arête](#) d'un [couplage](#) est dit **saturé**.

## Simple (simple)

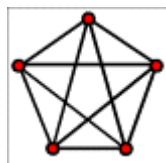
Un [graphe](#) est dit **simple**, s'il ne contient pas de [boucles](#) et s'il n'y a pas plus d'une [arête](#) reliant deux mêmes [sommets](#).

## Simplicial

Un [sommets](#) est dit **simplicial** si l'ensemble de ses [voisins](#) forme une [clique](#).

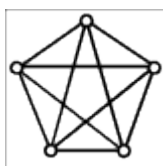
## Sommet (Vertex, pluriel Vertices)

Extrémité d'une [arête](#). Voici les sommets (en rouge) d'un graphe :

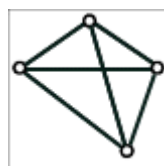


## Sous-graphe (Induced Subgraph)

Un sous-graphe d'un [graphe](#) est obtenu en y enlevant des [sommets](#) et toutes les [arêtes incidentes](#) à ces [sommets](#).



Un graphe



Un sous-graphe

## Stable (Stable)

Un stable d'un [graphe](#)  $G$  est un [sous-graphe](#) de  $G$  sans [arêtes](#). L'[ordre](#) du plus grand stable de  $G$  est noté  $\alpha(G)$  est s'appelle **nombre de stabilité**. Prononcer « alpha de  $G$  ».

## Taille (Size)

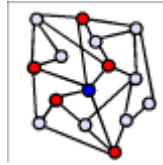
La taille d'un [graphe](#) est le nombre de ses [arêtes](#).

## Tournoi (Tournament)

[Digraphe complet](#).

## Voisinage (Neighborhood)

Le voisinage d'un [sommets](#) est l'ensemble de tous ses sommets [adjacents](#). Ci-dessous le voisinage (en rouge) du sommet bleu :



---

## Quelques références sur la théorie des graphes

### Livres

#### Pour débutants

- Cogis O., Robert Cl., **Théorie des graphes**, Vuibert, 2003
- Driesbeke F., Hallin M., Lefevre Cl., **Les graphes par l'exemple**, Ellipses, 1987
- Lipschutz Seymour, **Mathématiques discrètes**, Série Schaum, McGraw-Hill, 1990

#### Pour aller plus loin

- Gondran M., Minoux M., **Graphes et algorithmes**, 2e édition, Eyrolles, 1986
- Rüegg Alan, **Processus stochastiques**, Presses polytechniques romandes, 1989
- Skiena Steven, **Implementing Discrete Mathematics**, Addison-Wesley, 1990 (voir le site [www.combinatorica.com](http://www.combinatorica.com))

D'autres références sur le web sont disponibles à l'adresse : [www.apprendre-en-ligne.net/graphes/ref.html](http://www.apprendre-en-ligne.net/graphes/ref.html)

