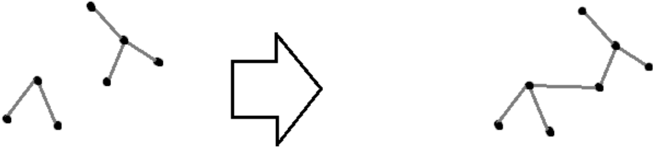
Chapitre III : Arbres



Deforestation:
When adding a branch gives you fewer trees.

Arbres : Introduction

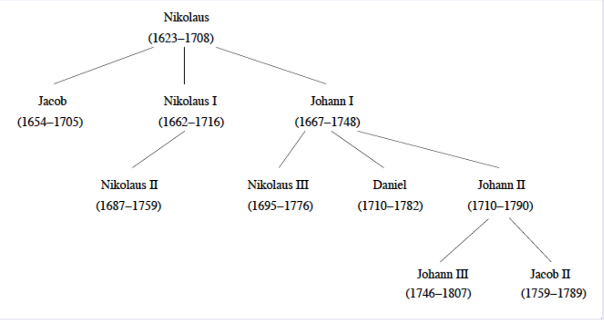
Un graphe (simple) connexe qui ne contient pas de circuits simples est appelé un arbre. Les arbres ont été utilisés dès 1857, quand le mathématicien anglais **Arthur Cayley** les utilisa pour compter certains types de composés chimiques. Depuis, les arbres ont été utilisés pour résoudre des problèmes une grande variété de disciplines.

Les arbres sont particulièrement utiles en informatique :

- Pour construire des algorithmes efficaces pour localiser des éléments dans une liste.
- Pour construire des réseaux reliant des ordinateurs distribués tout en optimisant le cout des cables optiques.
- Pour construire des codes efficaces pour stocker et transmettre les données.
- Pour modéliser des procédures qui sont effectuées en utilisant une séquence de décisions. Ce qui rend les arbres précieux dans l'étude des algorithmes de tri.

Exemple 1

Un arbre généalogique des ***Bernoullis****, une famille célèbre des mathématiciens suisses, est présenté ci-dessous. C'est un graphe où les sommets représentent les membres de la famille et les arêtes représentent les relations parent-enfant.*



Arbres : Introduction

Définition 1

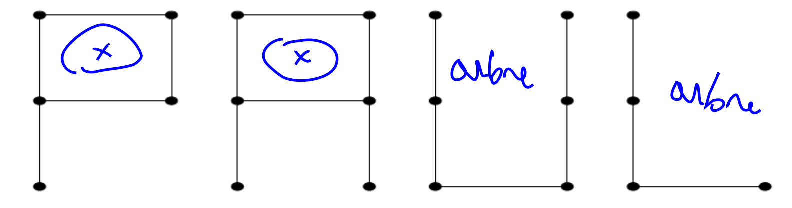
*Un **arbre** est un graphe non dirigé, acyclique (qui ne contient pas de cycles simples) et connexe.*

Remarque :
Puisqu'un arbre ne peut pas contenir de circuit simple, un arbre ne peut pas contenir d'arêtes multiples ni de boucles. Par conséquent, tout arbre doit être un **graphe simple**.

Arbres : Introduction

Exemple 2

Lesquels des graphes suivants sont des arbres ?



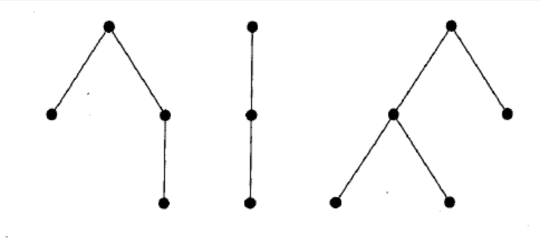
(a) (b) (c) (d)

Tout graphe connexe qui ne contient pas de circuits simples est un arbre. Qu'en est-il des graphes ne contenant pas de circuits simples qui ne sont pas nécessairement connexes ?

Définition 2

*Une **forêt** est un graphe simple non dirigé dont chaque composante connexe est un arbre, c'est en quelque sorte une "collection" d'arbres.*

Exemple 3



Arbres : Unique chemin simple

Les arbres sont souvent définis comme des graphes non dirigés avec la propriété qu'il existe un chemin unique entre chaque paire de sommets. Le théorème suivant montre que cette définition alternative est équivalente à notre définition.

Théorème 4

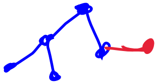
Un graphe non dirigé est un arbre si et seulement si il existe un unique chemin simple entre chaque paire de sommets.

Arbres : Unique chemin simple

Démonstration Montrons d'abord que si un graphe non dirigé est un arbre avec n sommets, alors il existe un chemin simple unique entre chaque paire de sommets par récurrence.
Soit $P(n)$ le fait que chaque arbre à n sommets contient un chemin simple unique entre chaque paire de sommets. On peut assument que $n \geq 2$.

- $P(2)$ est vraie, car entre 2 sommets il existe une arête unique, qui est le chemin recherché.
- On suppose que $P(n)$ est vrai pour $n \geq 2$.

Arbres : Unique chemin simple



- On montre que $P(n+1)$ est vrai.
  Prenons un arbre avec n sommets, auquel on rajoute 1 sommets et 1 arête. Cette nouvelle arête connecte le nouveau sommet à un sommet déjà existant. Ce qui crée un chemin simple entre le nouveau sommet et tout autre sommet de l'arbre. On ne peut rajouter d'autres arêtes sans contredire la condition de non-existence de circuit simple dans un arbre.

Par récurrence on a montré qu'un arbre à n sommets possède un chemin simple entre chaque paire de sommets.

Arbres : Unique chemin simple

Montrons à présent par contraposition que si un graphe possède un chemin simple unique entre chaque paire de sommets, alors ce graphe est un arbre.

Soit un graphe connexe, non-dirigé qui n'est pas un arbre. Alors il doit contenir un circuit simple (par définition). Donc pour toute paire de sommets contenu dans ce circuit simple, il existe (au moins) 2 chemins simple entre eux. Par conséquent, il n'existe pas de chemin simple unique entre chaque paire de sommets.

Par contraposition on à démontré que lorsqu'un graphe possède un chemin simple unique entre chaque paire de sommets, alors ce graphe est un arbre. □

Arbres : Unique chemin simple

Théorème 5

Chaque arbre est un graphe biparti.

<u>Démonstration</u> Fixons un sommet v_0. Soit V_1 l'ensemble de tous les sommets dont la longueur de chemin unique de v_0 est paire et soit V_2 l'ensemble de tous les sommets dont la longueur de chemin unique de v_0 est impaire.

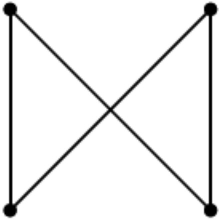
Ainsi, l'ensemble des sommets de l'arbre peut -être partitionné en deux sous-ensembles V_1 et V_2 de telle sorte que chaque arête de l'arbre relie un sommet de V_1 à un sommet de V_2.
Par conséquent, chaque arbre est un graphe biparti. $\square$

Arbres : Unique chemin simple

L'inverse de l'affirmation n'est pas vraie :

graphe biparti $\not\Rightarrow$ arbre

Contre-exemple :
Le graphe bipartite complet $K_{2,2}$ (illustré ci-dessous) constitue un contre exemple car il possède un circuit simple. Il existe également des graphes bipartis non connexes, ce qui est un autre moyen trouver des contre-exemples.



Arbres enracinés

Dans de nombreuses applications, un sommet particulier d'un arbre est désigné comme la racine. Une fois qu'une racine est désignée, nous pouvons assigner une direction à chaque arête comme suit. Etant donné qu'il existe un chemin unique entre la racine et chaque sommet de l'arbre (à partir du théorème 4 Chap 3), nous orientons chaque arête en partant de la racine.

Définition 3

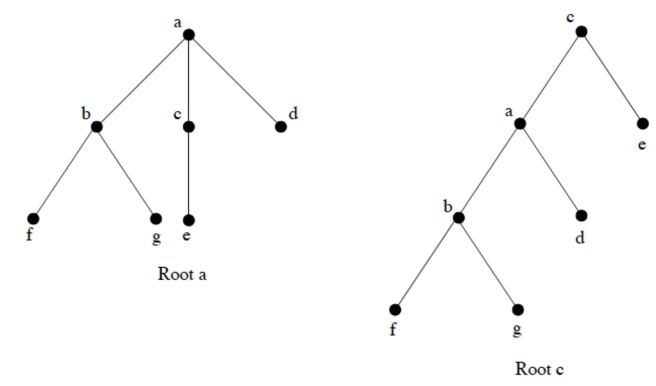
Un arbre avec sa racine génère un graphe dirigé appelé un ***arbre enraciné****.*

Nous pouvons transformer un arbre (sans racines) en arbre enraciné en choisissant un sommet comme étant la racine. Notez que différents choix de la racine produisent différents arbres enracinés.

Arbres enracinés

Exemple 6

La figure ci-dessous montre deux arbres enracinés formés en désignant respectivement a comme étant la racine et c comme étant la racine dans un arbre.



Remarque :
Nous dessinons habituellement un arbre enraciné avec sa racine tout en haut du graphe. Les flèches indiquant les directions des arêtes de l'arbre enraciné peuvent être omises, car le choix de la racine détermine les directions des arêtes.

Terminologies des arbres

La terminologie des arbres tient son origine de la botanique et la généalogie.

Définition 4

Soit T soit un arbre enraciné :

- *Si v est un sommet dans T autre que la racine, le **parent** de v est l'unique sommet u tel qu'il existe une arête dirigée de u vers v.*
- *Lorsque u est le parent de v, v est appelé un **enfant** de u.*
- *Les sommets avec le même parent sont appelés **frères (et soeurs)**.*

Terminologies des arbres

Définition 5

Soit T soit un arbre enraciné :

- *Les **ancêtres** d'un sommet différent de la racine sont tous les sommets dans le chemin de la racine à ce sommet, en excluant le sommet lui-même et en incluant la racine (c'est-à-dire son parent, le parent de son parent, etc. jusqu'à ce que la racine soit atteinte).*
- *Les **descendants** d'un sommet v sont tous les sommets qui ont v comme ancêtre.*
- *Le sommet d'un arbre est appelé **feuille (ou sommet terminal)** s'il n'a pas d'enfants.*

Terminologies des arbres

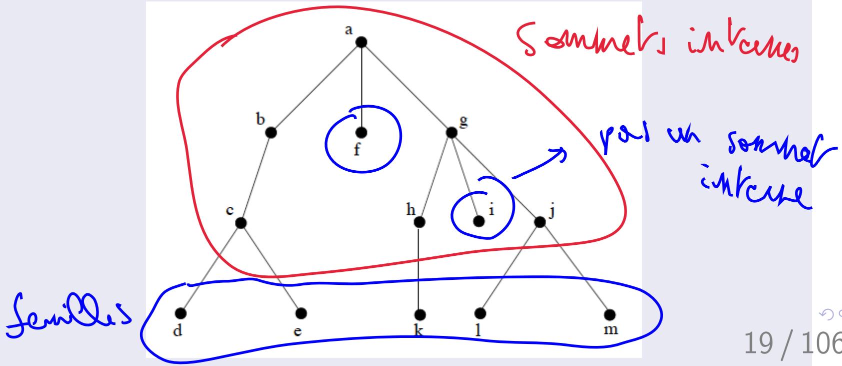
Définition 6

Soit T soit un arbre enraciné :

- *Les sommets ayant des enfants sont appelés **sommets internes**. La racine est un sommet interne à moins que ce soit le seul sommet du graphe, auquel cas il s'agit d'une feuille.*
- *Si a est un sommet dans un arbre, le **sous-arbre** avec a comme racine est le sous-graphe de l'arbre constitué du sommet a et tous ses descendants ainsi que toutes les arêtes incidentes à ces descendants.*

Exemple 7

Dans l'arbre enraciné T (avec la racine a) montré ci-dessous, trouver le parent de c, les enfants de g, les frères de h, tous les ancêtres de e, tous les descendants de b, tous les sommets internes et toutes les feuilles. Quel est le sous-arbre enraciné en g ?



Arbres m-aires et arbres binaires

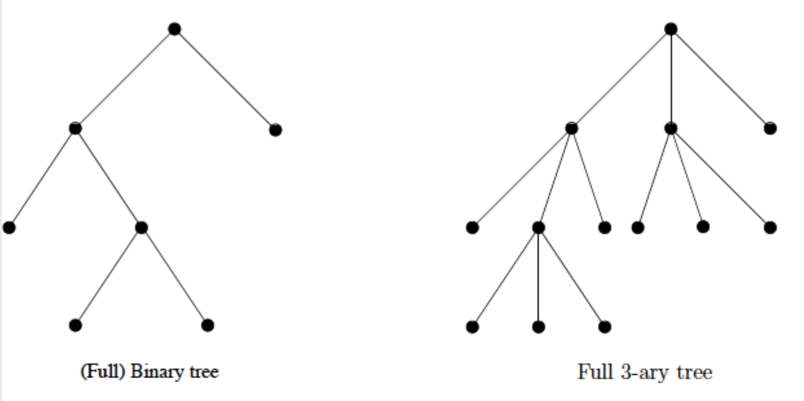
Les arbres enracinés avec la propriété que tous leurs sommets internes ont le même nombre d'enfants sont utilisés dans différentes applications. Comme par exemple les problèmes liés aux codes.

Définition 7

*Un arbre enraciné est appelé un **arbre** m**-aire** si chaque sommet à au plus m enfants. L'arbre est appelé un **arbre** m**-aire entier** si chaque sommet interne a exactement m enfants. Un arbre m-aire avec $m = 2$ est appelé un **arbre binaire**.*

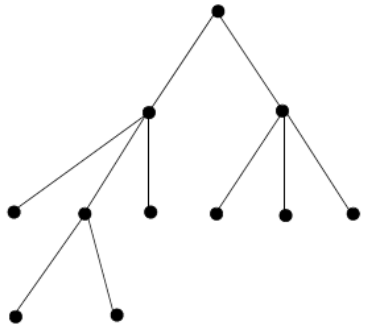
Arbres *m*-aires et arbres binaires

Exemple 8



Arbres m-aires et arbres binaires

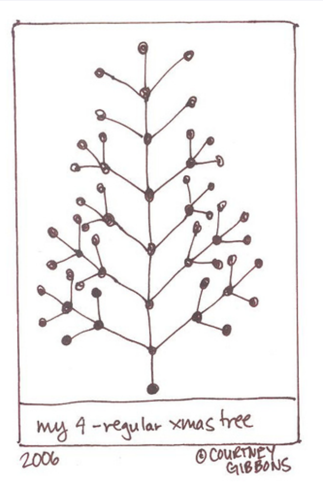
Exemple 9



3-ary tree

Arbres m-aires et arbres binaires

Exemple 10



Arbres enracinés ordonnés

Définition 8

*Un **arbre enraciné ordonné** est un arbre enraciné où les enfants de chaque sommet interne sont ordonnés. Les arbres enracinés ordonnés sont dessinés de telle sorte que les enfants de chaque sommet interne soient dans l'ordre de gauche à droite. Une représentation d'un arbre enraciné de manière conventionnelle détermine donc un ordre pour ses arêtes.*

Notons que cet arrangement des arêtes est utilisés dans les dessins sans mentionner explicitement que nous envisageons d'ordonner un arbre enraciné.

Arbres enracinés ordonnés

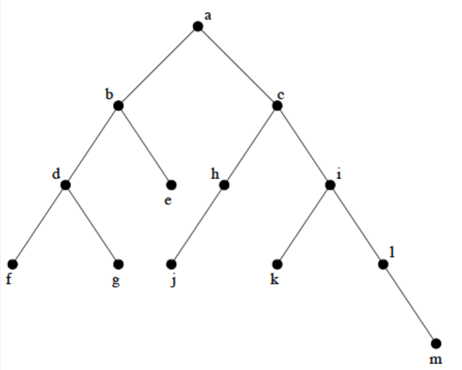
- Dans un arbre binaire ordonné, si un sommet interne a deux enfants, le premier enfant est appelé **l'enfant de gauche** et le deuxième enfant est appelé **l'enfant de droite**.
- L'arbre enraciné à l'enfant gauche d'un sommet est appelé le **sous-arbre gauche** de ce sommet et l'arbre enraciné à l'enfant de droite d'un sommet s'appelle le **sous-arbre de droite** du sommet.

Il faut noter que pour certaines applications, chaque le sommet d'un arbre binaire, autre que la racine, est désigné comme enfant droit ou gauche de son parent. Même lorsque certains sommets ont un seul enfant. Nous ferons ces désignations chaque fois que c'est nécessaire.

Arbres enracinés ordonnés

Exemple 11

Quels sont les enfants gauche et droit du sommet d dans l'arbre ordonné binaire T ci-dessous (où l'ordre est celui sous-entendu par le dessin)? Quels sont les sous-graphes gauche et droite du sommet c?



Arbres enracinés ordonnés

Tout comme pour les graphes, il n'existe pas de terminologie standard utilisé pour décrire les arbres, les arbres enracinés, arbres enracinés ordonnés et arbres binaires. Cette terminologie non standard vient du fait que les arbres sont utilisés très souvent en informatique, domaine relativement jeune.

Vérifier donc soigneusement le sens donné aux termes utilisés dés qu'ils apparaissent quelque part.

Arbres enracinés ordonnés

Étant donné un arbre enraciné ordonné, on peut considérer plusieurs méthodes pour lister tous les sommets de l'arbre.

Une approche s'appelle le **Preorder algorithm** : Preorder(T) runs on a tree T with root r by printing the root, then if $x_1, ..., x_k$ are the children (in order) the algorithm recursively calls Preorder(T_1), . . . , Preorder(T_k) where each T_i is the sub-tree with root x_i.

L'autre approche est le **Postorder algorithm** : Postorder(T) on a tree T with root r and children $x_1, ..., x_k$ in order runs Postorder (T_1), . . . , Postorder(T_k) and then prints the root r.

Applications des arbres

Les arbres sont utilisés comme modèles dans des domaines aussi divers que l'informatique, la chimie, la géologie, la botanique, la sociologie et la psychologie. Nous allons décrire quelques exemples.

Applications des arbres : Arbres en chimie

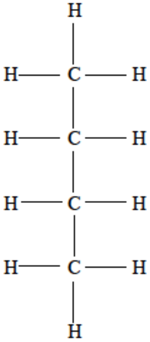
Une abstraction importante en chimie consiste à écrire des molécules sous forme de graphes. Les atomes sont représentés par les sommets et les liens chimiques sont décrits par les arêtes.
Parfois ces graphes ont des cycles (par exemple, le benzène) mais dans de nombreux cas importants ce sont des arbres.
Notez que, dans ces cas, ils sont naturellement des arbres sans racine !
Le mathématicien anglais **Arthur Cayley** a découvert des arbres en 1857 alors qu'il tentait dénombrer les isomères de composés de la forme C_nH_{2n+2}, appelés **hydrocarbures saturés**.

Applications des arbres : Arbres en chimie

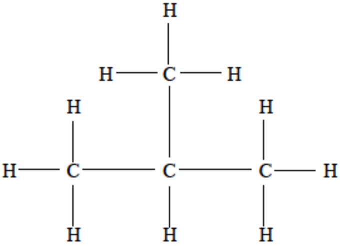
Dans les graphes d'hydrocarbures saturés, chaque atome de carbone est un sommet de degré 4 et chaque l'atome d'hydrogène est un sommet de degré 1. Il y a $3n+2$ sommets dans un graphe représentant C_nH_{2n+2}. Le nombre d'arêtes dans un tel graphe est la moitié de la somme des degrés des sommets. Par conséquent, il y a $(4n+2n+2)/2 = 3n+1$ arêtes dans ce graphe. Puisque le graphe est connexe et que le nombre d'arêtes est égale au nombre de sommets moins un (Théorème 11), il doit s'agir d'un arbre.

Les arbres non isomorphes (c'est-à-dire, ceux qui correspondent à des molécules différentes) à n sommets de degré 4 et $2n+2$ de degré 1 représentent les différents isomères de C_nH_{2n+2}.

Par exemple, lorsque $n = 4$, il y a exactement deux arbres non isomorphes de ce type (c'est un bon exercice de vérifier cela). Par conséquent, il y a exactement deux différents isomères de C_4H_{10}. Leurs structures sont ci-dessous.



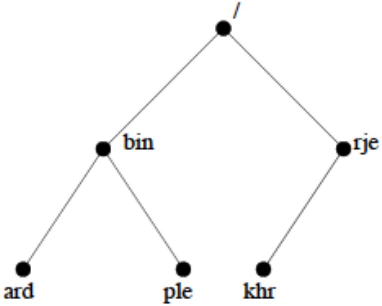
Butane



Isobutane

Applications des arbres : Système de fichiers informatiques

Les dossiers / répertoires d'un disque d'ordinateur ont une structure en arbre correspondant à des sous-dossiers / sous-répertoires. Le répertoire "racine" contient l'ensemble du système de fichiers.
Ainsi, un système de fichiers peut être représenté par un arbre enraciné, où la racine représente le répertoire racine, les sommets internes représentent des sous-répertoires, et les feuilles représentent des fichiers ou répertoires vides. Voici un exemple :



Propriétés des arbres : Compter les sommets et les arêtes

Nous aurons souvent besoin de résultats concernant le nombre d'arêtes et des sommets de différents types d'arbres.

Théorème 12

Un arbre à n sommets possède $n-1$ arêtes.

<u>Démonstration</u> par récurrence sur n.
Soit $P(n)$ le fait qu'un arbre à n sommets possède $n-1$ arêtes.

- $P(1)$ établit le fait qu'un arbre à 1 sommet n'a aucune arête ce qui est vrai. Si un graphe à 1 sommet possède une arête il en découle qu'il doit contenir un circuit simple ce qui est en contradiction avec la définition d'un arbre.
- Hypothèse : Soit $P(n)$ vrai pour un $n \geq 1$.

Propriétés des arbres : Compter les sommets et les arêtes

- Thèse : $P(n+1)$?
  Prenons un arbre à n sommets et $n-1$ arêtes, auquel on rajoute 1 sommet. Il faut dans ce cas rajouter également une arête sinon on perd la connectivité ce qui est en contradiction avec la définition d'un arbre. Selon le Théorème 4 on ne peut rajouter plus qu'une arête sans introduire un circuit simple dans le graphe, qui dés lors ne serait plus un arbre.
  Nous obtenons donc un arbre à $n+1$ sommets et $(n-1)+1$ arêtes. Ce qui implique que $P(n+1)$ est vrai.

Par récurrence nous avons montré que $P(n)$ est vrai pour tout $n \geq 1$. $\square$

Propriétés des arbres : Compter les sommets et les arêtes

Le nombre de sommets dans un arbre m-aire entier est déterminé par le nombre de sommets internes. Nous utiliserons n pour désigner le nombre de sommets dans un arbre.

Théorème 13

Un arbre entier avec i sommets internes contient $n = mi + 1$ sommets.

Démonstration par récurrence sur i.
Soit $P(i)$ le fait qu'un arbre m-aire entier avec i sommets internes contient $n = mi + 1$ sommets. On peut assumer que $i \geq 0$.

- $P(0)$ est le fait qu'un arbre entier m-aire avec aucun sommet interne à 1 sommet (n=m.0+ 1) et donc aucune arête. Ce qui est vrai.
- Hypothèse : Soit $P(i)$ vrai pour un $i \geq 0$.

Propriétés des arbres : Compter les sommets et les arêtes

- Thèse : $P(i+1)$?
  Soit un arbre avec i sommets interne, qui est m-aire et entier. On va changer un des sommets terminaux en sommet interne. Nous obtenons maintenant un arbre m-aire entier avec $i+1$ sommets internes. On obtient :

  $$n = mi + 1 + m = m(i+1) + 1$$

  Donc $P(i+1)$ est vrai.

Par récurrence nous avons montré que $P(i)$ est vrai pour tout $i \geq 0$. $\square$

Propriétés des arbres : Compter les sommets et les arêtes

Exemple 14

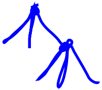
Soit T un arbre m-aire entier comportant 28 sommets. Quelles sont les valeurs possibles pour m ?

$28 = n = mi + 1$ $\quad mi = 27 \rightarrow m = \underline{1, 3, 9, 27}$

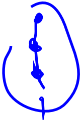
$m = 27$

$m = 9$

$m = 3$



$m = 1$



Propriétés des arbres : Compter les feuilles

Supposons que T soit un arbre m-aire entier à n sommets. Soit i le nombre de sommets internes et l le nombre de feuilles. Une fois que l'un des n, i et l est connu, les deux autres quantités sont déterminées. Comment les trouver ?

Propriétés des arbres : Compter les feuilles

Théorème 15

Un arbre m-aire entier

1. *à n sommets a*
   - $i = \dfrac{n-1}{m}$ *sommets internes et*
   - $l = \dfrac{(m-1)n+1}{m}$ *feuilles,*
2. *à i sommets internes a*
   - $n = mi + 1$ *sommets et*
   - $l = (m-1)i + 1$ *feuilles,*
3. *avec l feuilles a*
   - $n = \dfrac{ml-1}{m-1}$ *sommets et*
   - $i = \dfrac{l-1}{m-1}$ *sommets internes.*

$i = \frac{n-1}{m}$

Démonstration Notons que $n = i + l$.

1.
   - Selon le Théorème 12 on sait que $n = m.i + 1$, il suffit d'isoler i.
   - Insérer l'équation précédente dans $l = n - i$. $= \frac{nm}{n}\left(\frac{n-1}{m}\right) - \frac{m(n-1)+1}{m}$

2.
   - Théorème 12
   - Il suffit d'utiliser $l = n - i$ et l'équation précédente.

3.
   - Isoler n dans l'équation suivante $n - l = i = \dfrac{n-1}{m}$
   - Insérer l'équation précédente dans $i = n - l$. □

arbre 4-aire entier

$\ell = 100$

Exemple 16

Supposons que quelqu'un commence une chaine par lettre. Chaque personne qui reçoit la lettre est priée de l'envoyer le à quatre autres personnes. Certaines personnes le font, mais d'autres n'envoie aucune lettre. Combien de personnes ont vu la lettre, y compris la première personne, si personne ne reçoit plus d'une lettre et si la chaîne se termine après que 100 personnes l'aient lu mais ne l'ont pas envoyé ? Combien de personnes ont envoyé la lettre ?

$m = 4$

$\ell = 100$

$n = \dfrac{m\ell - 1}{m - 1} = 133$

i

$i = \dfrac{\ell - 1}{m - 1} = 33$

Propriétés des arbres : Hauteur et niveaux

Pour de nombreuses applications, il est souhaitable d'utiliser des arbres enracinés qui sont "**équilibrés**", de sorte que les sous-arbres à chaque sommet contiennent des chemins d'environ la même longueur. Le concept sera plus clair après quelques définitions.

Définition 9

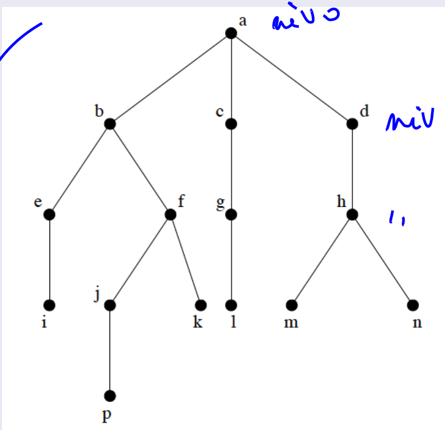
- *Le **niveau** d'un sommet v dans un arbre enraciné est la longueur du chemin unique de la racine à ce sommet. Le niveau de la racine est défini à zéro.*

- *La **hauteur** d'un arbre enraciné est le maximum des niveaux des sommets. En d'autres termes, la hauteur d'un arbre enraciné est la longueur du plus long chemin de la racine à n'importe quel sommet.*

Exemple 17

Trouver le niveau de chaque sommet dans l'arbre enraciné ci-dessous. Quelle est la hauteur de cet arbre ?

h= 4



niv 0

niv 1

" 2

Exemple 18

Soit T un arbre m-aire entier comportant 50 sommets. Que doit être m si la hauteur de T est 2 ?

$n = mi + 1 \to i = \frac{50-1}{m}$

Solution :

$i = 49/m$. *Comme i est un entier alors m doit être 1, 7 ou 49.*

- *Si $m = 1$, alors un arbre de hauteur 2 à exactement 3 sommets.*
- *Si $m = 7$, alors un arbre de hauteur 2 avec 50 sommets peut exister (exercice).*
- *Si $m = 49$, alors T n'a qu'un seul sommet interne et donc est de longueur 1.*

Donc si T est de hauteur 2, alors $m = 7$.

Définition 10

*Un arbre m-aire enraciné de hauteur h est appelé **équilibré** si toutes les feuilles sont aux niveaux h ou h − 1.*

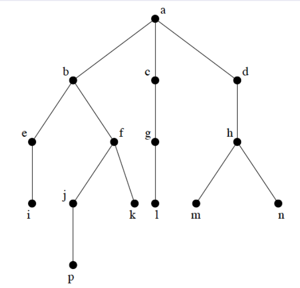
Exemple 19

L'arbre enracine montré ci-dessous est-il équilibré ? Dessiner un arbre ce n'est pas équilibré. ↳ oui

niv 1

niv 3

↳ pas ép.



Les résultats suivants concernent la hauteur et le nombre de feuilles dans les arbres m-aires.

Théorème 20

Il y a tout au plus m^h feuilles dans un arbre m-aire de hauteur h.

Démonstration par récurrence sur h.
Soit $P(h)$ le fait qu'il y a au plus m^h feuilles dans un arbre m-aire de hauteur h. On peut assumer que $h \geq 0$.

- $P(0)$ est le fait qu'il y a au plus $m^0 = 1$ feuille dans un arbre m-aire de hauteur 0. Ce qui est vrai.
- Hypothèse : Soit $P(h)$ vrai pour un $h \geq 0$.

- Thèse : $P(h+1)$?
  Soit un arbre m-aire de hauteur h. On augmente la hauteur de l'arbre de 1, toutes les feuilles deviennent des sommets internes en rajoutent m enfants à chacun d'entre eux.
  Ceci nous donne un arbre m-aire de hauteur $h+1$ avec au plus $m^h.m = m^{h+1}$ feuilles. Donc $P(h+1)$ est vrai.

Par récurrence nous avons montré que $P(h)$ est vrai pour tout $i \geq 0$. $\square$

Propriétés des arbres : Equilibre

Corollaire 21

- *Si un arbre m-aire de hauteur h a l feuilles, alors* $h \geq \lceil \log_m l \rceil$.
- *Si un arbre m-aire est entier et équilibré alors* $h = \lceil \log_m l \rceil$.

(Fonction ceil : $\lceil x \rceil$ est le plus petit entier plus grand ou égale à x.)

Nous ne donnons pas la démonstration ici mais elle se base sur :
$l \leq m^h$ en prenant le logarithme en base m (≥ 1) de chaque coté de l'équation et ensuite la fonction ceil.

Arbres d'expressions : expressions arithmétiques

Les expressions mathématiques composées d'entiers et de variables entières combinées par addition, soustraction et multiplication, sont appelées des **expressions arithmétiques**.

Quelques exemples :
$0, 2, x, y, 2 + x, x.y, ((x.y) - (2 + x))$ et $((x.y) - (2 + x)) + y$

Les expressions arithmétiques peuvent être définies de manière inductive comme suit :

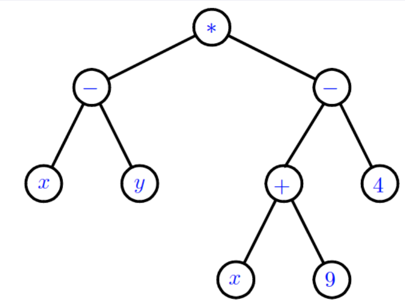
- Les variables entières et les entiers sont des expressions arithmétiques.
- Si A et B sont des expressions arithmétiques, alors il en est de même pour $A + B, A - B$ et $A \star B (= A.B)$.

Arbres d'expressions

Chaque expression arithmétique peut être représentée par un arbre, appelé un **arbre d'expression**, utilisés avec "l'arbre de formation" d'une formule en logique propositionnelle.

Exemple 22

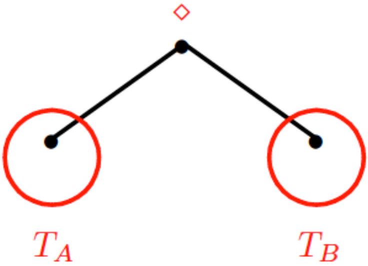
$(x - y) \star ((x + 9) - 4)$ *peut être représenté par*



Arbres d'expressions : Définition inductive

Soit T_e l'arbre représentant l'expression arithmétique e.

- Base : Si e est un entier ou une variable entière, prenons T_e comme un sommet unique étiqueté e.
- Récurrence : Si e est de la forme $A \diamond B$ (où $\diamond$ est $+, -$ ou $.$), et les arbres d'expression T_A et T_B (pour A et B), et créer T_e en introduisant une nouvelle racine avec l'étiquette $\diamond$, et des nouvelles arêtes de ce sommet aux racines de T_A et T_B.



Arbres d'expressions : Définition inductive

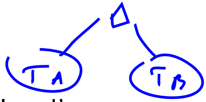
Théorème 23

Dans un arbre d'expression T, chaque feuille est étiquetée avec un entier ou une variable entière, et chaque sommet intérieur est étiqueté avec $+, .$ *ou* $-$.

Démonstration Nous utilisons la récurrence sur le nombre de sommets (n).

- Etape de base : Si T a un sommet, l'expression est un seul entier ou une variable entière, et ce sommet est une feuille (et il n'y a pas de sommets intérieurs), donc la conclusion tient.

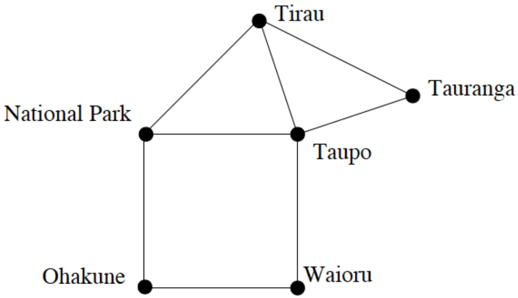
Arbres d'expressions : Définition inductive



- Récurrence : Supposons que T ait plus d'un sommet. Alors T est l'arbre pour une expression de la forme $A \diamond B$, avec la racine étiquetée $\diamond$.
  Par induction, on peut supposer que T_A et T_B ont les propriétés requises.
  Maintenant chaque feuille de T est une feuille de T_A ou de T_B, et est donc étiqueté avec un entier ou variable entière, alors que chaque sommet intérieur de T autre que la racine est un sommet intérieur de T_A ou T_B étiqueté par conséquent par $+, .$ ou $-$. $\square$

Arbres sous-tendants

Considérons le système de routes dans l'île du Nord représenté par le graphe simple connexe ci-dessous. S'il neige, la seule façon de garder une route ouverte est d'utiliser un chasse-neige, mais cela coûte beaucoup d'argent. Le département des transports veut dégager le moins de routes possible afin qu'il soit possible de rouler entre n'importe quelle paire de villes. Comment cela peut-il être fait ?



Arbres sous-tendants

Au moins cinq routes doivent être déneigées pour garantir un chemin entre chaque paire de villes. Notons que le sous-graphe représentant ces routes est un arbre, car il est connexe et contient six sommets et cinq arêtes.

Ce problème a été résolu en utilisant un sous-graphe connexe avec le nombre minimum d'arêtes et contenant tous les sommets du graphe simple d'origine. Un tel graphe doit être un arbre.

Arbres sous-tendants

Définition 11

Soit G un graphe simple. Un arbre ***sous-tendant*** *de G est un sous-graphe de G qui est un arbre et qui contient chaque sommet de G.*

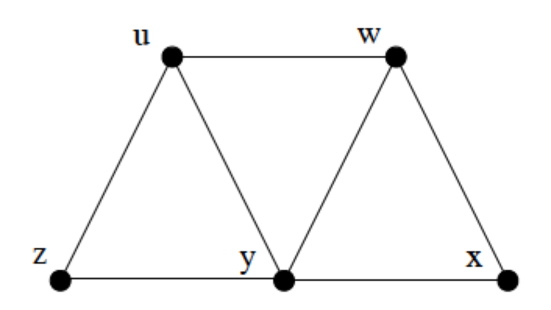
Notons qu'un arbre sous-tendant n'est donc PAS unique.

Un graphe simple avec un arbre sous-tendant doit être connexe, car il existe un chemin dans l'arbre sous-tendant entre toute paire de sommets. L'inverse est également vrai ; c'est-à-dire chaque graphe simple connexe a au moins un arbre sous-tendant. Nous allons donner un exemple avant de démontrer ce résultat.

Arbres sous-tendants

Exemple 24

Trouver un arbre sous-tendant du graphe simple G ci-dessous.



Arbres sous-tendants

Une autre caractérisation des graphes connexe est la suivante :

Théorème 25

Un graphe simple est connexe si et seulement si il possède un arbre sous-tendant.

Si un arbre est l'arbre sous-tendant d'un graphe simple, alors le graphe simple est connexe. La démonstration dans ce sens ne sera pas donnée ici mais elle se base sur le fait que tout arbre est un graphe simple et connexe.

Arbres sous-tendants

Démonstration Si un graphe simple G est connexe alors il possède un arbre sous-tendant.

Supposons que G est un graphe simple et connexe. Si

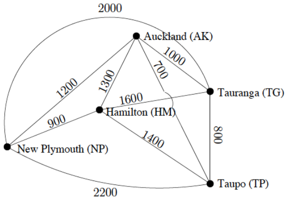
- G ne contient pas de circuit simple, alors G est son propre arbre sous-tendant.
- G contient des circuits simples (un nombre fini), alors on retire une arête sans perdre la connectivité du graphe qui devient le graphe G'. Si
  - G' ne contient pas de circuit simple, alors G' est son propre arbre sous-tendant.
  - G' contient des circuits simples (un nombre fini), alors on retire une arête sans perdre la connectivité du graphe qui devient le graphe G''.

Comme G est un graphe fini il suffit de répéter le processus un nombre fini de fois avant de trouver l'arbre sous-tendant pour G. □

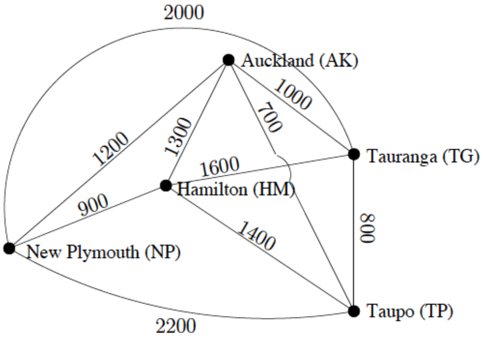
Exemple 26

Une entreprise souhaite mettre en réseau ses cinq centres de calculs. Toute paire de centres peut être reliée par des cables de fibre optique. Quels liens doivent être établis pour assurer qu'il y a un chemin entre deux centres informatiques tout en minimisant le coût total du réseau ?

On modélise avec un graphe pondéré, les sommets représentent les centres de calculs, les arêtes représentent les possibilités de câbles et les poids sur les arêtes sont les taux de location mensuels des lignes.



On peut résoudre ce problème en trouvant l'arbre sous-tendant de telle sorte que la somme des poids des arêtes de l'arbre est minimisée. Un tel arbre est appelé un arbre sous-tendant minimal.



Arbres sous-tendants minimaux

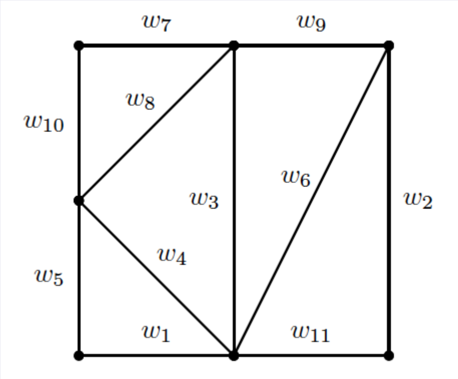
Une grande variété de problèmes peuvent être résolus en trouvant un trouvant les arbres sous-tendants dans un graphe pondéré tel que la somme des poids des arêtes dans l'arbre minimal.

Définition 12

*Un **arbre sous-tendant minimal** dans un graphe connexe pondéré est un arbre sous-tendant qui, par rapport à tous autres arbres sous-tendant possibles, a la plus petit possible somme des poids de ses arêtes.*

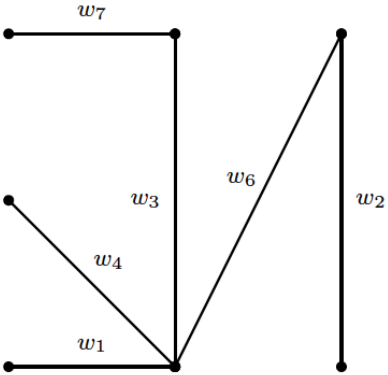
Exemple 27

Soit G le graphe connexe pondéré ci-dessous ou $w_i < w_j$ pour $i < j$. Dessiner l'arbre sous-tendant minimal pour G.



Arbres sous-tendants minimaux

Solution :



Arbres sous-tendants : Algorithmes

Nous mentionons deux algorithmes pour construire un arbre sous-tendant minimal (description complète dans un autre cours).

Les deux procèdent en ajoutant successivement des arêtes de plus petit poids. Ces algorithmes sont des exemples d'**algorithmes "gloutons"**. Un algorithme glouton est une procédure qui fait un choix optimal à chacun de ses pas. L'optimisation à chaque étape ne garanti pas que la solution globale optimale est produite.

Les deux algorithmes mentionnés sont des algorithmes gloutons qui produisent des solutions optimales :

- Algorithm de Prim → ajouter des arêtes "poids min" adj au sommets existants
- Algorithm de Kruskal → ajouter arêtes poids min pour créer chemin simple

Définition 13

*Un arbre à n sommets dont chaque sommet est étiqueté par une étiquette allant de 1 à n est un **arbre étiqueté**.*

Les **arbres étiquetés** (non planaires) sont parfois appelés arbres de Cayley.

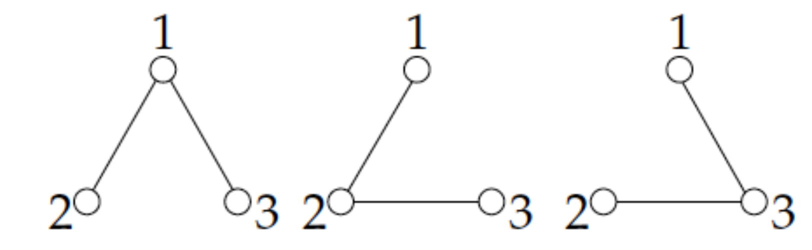
Parfois on peut compter le nombre d'éléments d'un ensemble A en trouvant une bijection entre A et un autre ensemble B dont on connaît le nombre d'éléments. Alors évidemment, $|A| = |B|$.

Théorème 28 (Théorème de Cayley, 1889)

Le nombre d'arbres étiquetés à n sommets est exactement n^{n-2}.

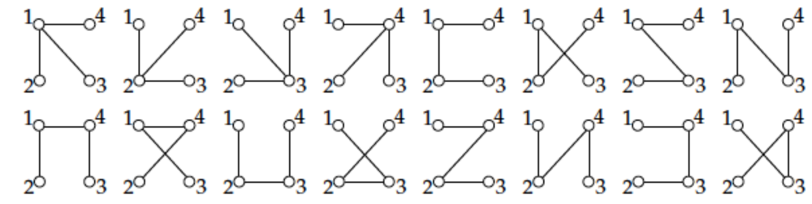
Exemple 29

pour $n = 3$, on trouve les $3 = 3^{3-2}$ arbres suivants.



Exemple 30

Pour $n = 4$, on trouve les $16 = 4^{4-2}$ arbres suivants.



Th Cayley $\left| \begin{array}{c} \text{arbres étiquetés} \\ n \text{ sommets} \end{array} \right| = n^{n-2}$

Soit a_n = # arbres étiq. à n sommets

A_n = l'ensemble des arbres étiq à

n sommets 2 sommets spéciaux

○ extrémité gauche

□ ... droite

(peuvent coïncider)

$|A_n| = n^2 \cdot a_n$

But : $|A_n| = n^n$

$[n]^{[n]}$ $\quad [n] = \{1, \dots, n\}$

$\rightarrow \exists$ bij $A_n \rightarrow [n]$ fcts

Soit $f : [n] \rightarrow [n]$ $\quad G(f)$ = graphe dirigé

sommets $1, \dots, n$

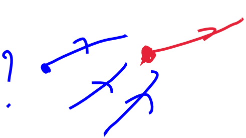
arêtes $(i, f(i))$

$i \in \{1, \dots, n\}$

$G(f)$ = n arêtes

= chaque sommet est relié (= chaque sommet a une image)

$\Rightarrow \exists!$ arête sortante à chaque sommet (déf d'une fct.)



$(0, \ldots, n)$

Dans chaque composante connexe $G(\vec{f})$ cycle dirigé

$M = \{$ sommets de $G(\vec{f})$ dans le cycle dirigé $\}$

= le + grand sous-ensemble de $[n]$ sur lequel f est bij.

Posons $M = \{i_1, \dots, i_m\}$ $i_1 < i_2 < \dots < i_m$

$$f|_M = \begin{pmatrix} i_1 & \dots & i_m \\ f(i_1) & \dots & f(i_m) \end{pmatrix}$$

$f|_M : M \to M$ bijectif

$\to$ ts les élém. de M apparaissent 1 fois sur la 2e ligne

$$\sigma = (f(i_1), \dots, f(i_m)) = \square$$

Pour le reste, on utilise les mêmes de $F(\vec{f})$

on enlève l'orientation non dirigé

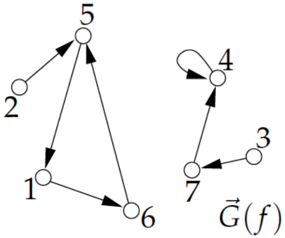
On obtient un arbre étiqueté n sommets

2 sommets 0, D

$\Rightarrow$ 1 élément à l'ensemble A_n

Afin de démontrer le Théorème de Cayley nous aurons besoin de la notation suivante :

Par exemple, pour $n = 7$ et $f = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 6 & 5 & 7 & 4 & 1 & 5 & 4 \end{pmatrix}$ on a le graphe suivant.



Ici, la fonction f est écrite comme une matrice $2 \times n$. Chaque colonne donne en deuxième ligne l'image de l'élément correspondant en première ligne.

Démonstration du théorème de Cayley. Soit a_n le nombre d'arbres étiquetés à n sommets numérotés de 1 à n, et soit $\mathcal{A}_n$ l'ensemble des arbres étiquetés à n sommets numérotés de 1 à n, avec deux sommets spéciaux $\bigcirc$ et $\square$ ($\bigcirc$ et $\square$ peuvent coincider). $\bigcirc$ est appelé **extrémité gauche** et $\square$ est appelé **extrémité droite**.
Remarquons qu'on a

$$|\mathcal{A}_n| = n^2 a_n$$

car pour tout arbre à n sommets il y a n choix pour $\bigcirc$ et n choix pour $\square$. Notre but est de démontrer $|\mathcal{A}_n| = n^n$ en donnant une bijection entre $\mathcal{A}_n$ et $[n]^{[n]}$, l'ensemble des fonctions de $[n]$ dans $[n]$.
Soit $f : [n] \to [n]$ une fonction et soit $\vec{G}(f)$ le graphe dirigé dont les sommets sont les nombres 1, ..., n et les arêtes sont les couples $(i, f(i))$ avec $i \in [n]$ (chaque sommet est lié par une arête vers son image).

Le graphe $\vec{G}(f)$ possède exactement n arêtes. De chaque sommet sort exactement une arête (voyez-vous pourquoi ?). Le nombre d'arêtes entrantes en un sommet varie de 0 à n. On peut voir que chaque composante connexe de $\vec{G}(f)$ contient un unique cycle dirigé.

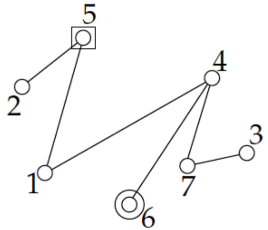
On définit $\mathcal{M}$ comme l'ensemble des sommets du graphe $\vec{G}(f)$ qui apparaissent sur un cycle dirigé (il peut y avoir plusieurs tels cycles dirigés). Intuitivement, ce sont ces cycles dirigés qui empêchent $\vec{G}(f)$ de définir un arbre. De manière équivalente, on pourrait définir $\mathcal{M}$ comme le plus grand sous-ensemble de $[n]$ sur lequel f est bijective.

Posons $\mathcal{M} := \{i_1, i_2, \ldots, i_m\}$ avec $i_1 < i_2 < \cdots < i_m$ et écrivons la restriction de f à $\mathcal{M}$ comme

$$f|_{\mathcal{M}} = \begin{pmatrix} i_1 & i_2 & \cdots & i_m \\ f(i_1) & f(i_2) & \cdots & f(i_m) \end{pmatrix}.$$

Pour rappel, la seconde ligne de cette représentation de $f|_{\mathcal{M}}$ (la restriction de f à $\mathcal{M}$) donne le vecteur des images de $i_1, i_2, \ldots, i_m$. Par définition, $f|_{\mathcal{M}}$ est un bijection de $\mathcal{M}$ vers $\mathcal{M}$, donc tous les éléments de $\mathcal{M}$ apparaissent dans la seconde ligne, dans un certain ordre.
L'idée maintenant est de représenter ce vecteur des images par un chemin dont les sommets sont $\bigcirc = f(i_1), f(i_2), \ldots, f(i_m) = \square$ (et ainsi se débarrasser des cycles). Pour le reste on utilise les arêtes de $\vec{G}(f)$ dont on enlève l'orientation (pour obtenir des arêtes, qui par définition n'ont pas de direction). On obtient ainsi un arbre étiqueté à n sommets dont deux sont distingués, c'est-à-dire un élément de $\mathcal{A}_n$.

Pour l'exemple précédent, avec $n = 7$ et
$f = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 6 & 5 & 7 & 4 & 1 & 5 & 4 \end{pmatrix}$, on a $\mathcal{M} = \{1, 4, 5, 6\}$ et
$f|_{\mathcal{M}} = \begin{pmatrix} 1 & 4 & 5 & 6 \\ 6 & 4 & 1 & 5 \end{pmatrix}$. Donc $\bigcirc = 6$ (l'extrémité "gauche") et $\square = 5$ (l'extrémité "droite"). Il faut remplacer les cycles dirigés 1–6–5–1 et 4–4 par le chemin 6–4–1–5. L'arbre obtenu est représenté ci-dessous.



1 zu 1 $[n]^{[n]} \to A_n$

$f \to \exists!$ élém. à A_n

bij

$\Rightarrow$ bij $[n]^{[n]} \leftrightarrow A_n$

$\Rightarrow |A_n| = n^n$

$\Rightarrow a_n = \frac{n^n}{n^2} = n^{n-2}$

obtenu

$|A_n| = n^2 a_n$

$\boxtimes$

Ceci définit une correspondance (c'est-à-dire une fonction) de $[n]^{[n]}$ vers $\mathcal{A}_n$. A chaque fonction f on fait correspondre exactement un élément de $\mathcal{A}_n$. Cette correspondance est bijective car il existe une correspondance réciproque associant à tout élément de $\mathcal{A}_n$ une fonction f, telle qu'effectuer les deux correspondances l'une après l'autre donne toujours l'identité. La correspondance de départ est une bijection de $[n]^{[n]}$ vers $\mathcal{A}_n$.

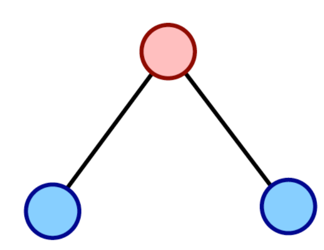
En conclusion on obtient $n^2 a_n = \mid \mathcal{A}_n \mid = n^n$ l'équation $a_n = n^{n-2}$. $\square$

Voyons maintenant un autre exemple de preuve bijective, à propos d'arbres binaires enracinés et triangulations d'un polygone. On l'utilisera par la suite pour obtenir une formule pour le **nombre de triangulations d'un polygone à $n+1$ côtés**, à partir d'une formule pour le **nombre d'arbres binaires entier enracinés à n feuilles**.

Bien que ces formules ne seront établies que plus tard, le Théorème 34 implique que dès qu'on sait compter une sorte d'objets, on sait automatiquement compter l'autre, vu que ensembles correspondants sont en bijection.

Exemple 31

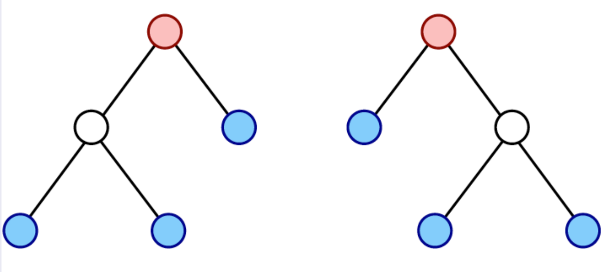
Il y a 1 *arbre binaire entier enraciné à* $n = 2$ *feuilles.*



Arbres et énumérations : Preuves bijectives

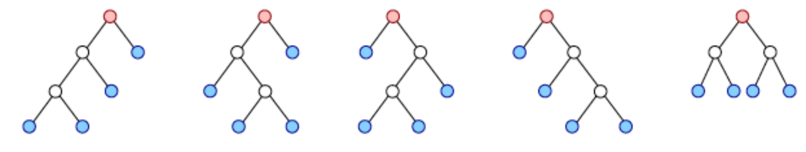
Exemple 32

Il y a 2 arbres binaires entier enracinés à $n = 3$ feuilles.



Exemple 33

Pour $n = 4$ feuilles, il y en a 5.



Attention : Quand on compte des objets, il faut être bien précis sur ce qu'on compte réellement !

En particulier, sur la manière de décider quand deux objets sont considérés comme égaux, et quand deux objets sont considérés comme différents. Rappelons nous de la définition formelle de ce qu'est un arbre enraciné.

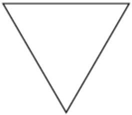
Un arbre binaire enraciné entier est un arbre (c'est-à-dire un graphe connexe, sans cycle) dont tous les sommets ont pour degré $1, 2$ ou 3. Un et un seul sommet a pour degré 2, la racine. Les sommets de degré 1 sont appelés feuilles. Les autres sommets sont les sommets internes de l'arbre. Chaque arête porte un label "G" ou "D" (pour "gauche" et "droite").

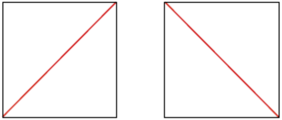
On considère deux arbres binaires comme égaux si l'on peut les superposer l'un sur l'autre sans se préoccuper du nom des sommets, mais en se préoccupant des labels. Ceci n'est évidemment pas une définition formelle. Pour éviter un long détour, disons simplement qu'il suffit de se souvenir de la notion isomorphisme d'arbres binaires enracinés entier.

Le problème qui nous intéresse est de compter les arbres binaires enracinés entiers à n feuilles, à isomorphisme près, pour un n donné.

Nous ne pouvons pas encore résoudre ce problème (ce sera fait plus tard, au chapitre 5. Par contre, nous pouvons d'ores et déjà faire une observation intéressante :

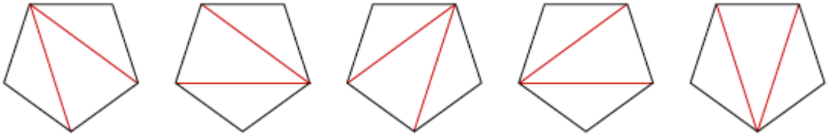
Comptons le nombre de triangulations d'un polygone à $n+1$ côtés.
Pour $n=2$, on trouve 1 triangulation.





Pour $n=3$, on en trouve 2.

Pour $n = 4$, on en trouve 5.



Ces nombres coïncident précisément avec le nombre d'arbres binaires enracinés entiers à n feuilles (comptés à isomorphisme près) pour $n = 2, 3, 4$. Ceci n'est pas un hasard, mais le signe d'un phénomène plus général.

Théorème 34

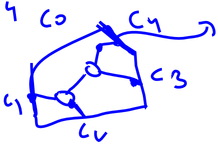
Le nombre d'arbres binaires enracinés entiers à n feuilles (à isomorphisme près) est égal au nombre de triangulations d'un polygone à $n+1$ côtés.

Avant de donner une idée de la preuve, voyons ce que signifie ce théorème. On ne sait pas (encore) comment compter les arbres enracinés à n feuilles (à isomorphisme près), ni les triangulations d'un polygone à $n+1$ côtés. Par contre, ce que le Théorème 34 montre, c'est que les nombres correspondants sont égaux, et ce pour toute valeur de n ! Dès qu'on pourra compter les objets d'un type, on pourra automatiquement compter les objets de l'autre type.

Idée de la démonstration :
Nous allons fournir une bijection entre l'ensemble $\mathcal{A}_n$ des arbres binaires entier enracinés à n feuilles, pris à un isomorphisme près, et l'ensemble $\mathcal{T}_{n+1}$ des triangulations d'un $(n+1)$-gone.
Numérotons les côtés d'un $(n+1)$-gone, disons P, de c_0 à c_n dans le sens anti-horlogique (dans la figure slide 88, le coté c_0 est représentée en haut).
Etant donné une triangulation de P, nous obtenons un arbre binaire enraciné entier à n feuilles en plaçant :

- la racine dans l'intérieur du triangle adjacent du côté c_0,
- un sommet (interne) dans l'intérieur de chaque autre triangle,
- les n feuilles dans l'intérieur (relatif) des côtés correspondants, de c_1 jusque c_n,

$n = 4$



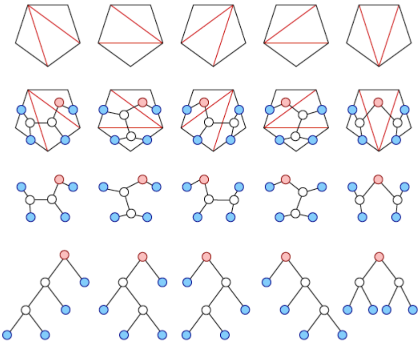
racine

● = feuilles

○ = sommets internes

puis en reliant les paires de sommets placés dans des triangles adjacents, ainsi que chaque feuille avec le sommet placé dans le triangle contenant le côté correspondant.

Et pour les labels ? On utilise l'orientation du plan. Ces règles définissent une application f de $\mathcal{T}_{n+1}$ dans $\mathcal{A}_n$, qui est illustrée ci-dessous (un dessin vaut parfois mieux qu'un long discours).



Pour compléter la démonstration on démontre que cette application f admet une application g de $\mathcal{A}_n$ dans $\mathcal{T}_{n+1}$ telle que $f \circ g$ et $g \circ f$ sont l'identité sur leurs domaines respectifs. Par le Théorème 7 (du chapitre 1), f est une bijection. $\square$

Tout arbre (fini) $T = (V, E)$ possède au moins une feuille, c'est-à-dire un sommet de degré 1 (à condition que T ait au moins deux sommets). Si on retire cette feuille, on trouve un nouvel arbre T' avec un sommet de moins. Si n désigne le nombre de sommets de l'arbre T de départ, alors T' a $n - 1$ sommets. En continuant d'**effeuiller** ainsi notre arbre, on aboutira nécessairement à un arbre ayant un seul sommet.

Chaque fois qu'une feuille de l'arbre courant est retirée, une et une seule arête disparaît de l'arbre. Le nombre d'arêtes de T est donc exactement $n - 1$, le nombre de sommets retirés avant de terminer avec un seul sommet.

C'est une autre manière d'arriver à une conclusion qu'on connait déjà pour les arbres qui est : $|E| = n - 1$.

Remarque : [Code de Prüfer]
Le processus d'effeuillage décrit au slide précedent donne lieu au **code de Prüfer** d'un arbre, un vecteur à $n-2$ composantes dont la $i$ème composante est l'étiquette du *voisin* du $i$ème sommet enlevé dans l'arbre courant.

Quelques précisions :

1. A chaque étape on choisit la feuille qui a la plus petite étiquette et
   - on écrit son "voisin" dans le code ;
   - on enlève cette feuille de l'arbre qui devient un sous-arbre ;
2. On arrête d'écrire le code dès qu'il ne reste plus que deux sommets ($=K_2$).

Pour fixer les idées, prenons $V = [n]$. Alors le code de Prüfer de l'arbre T est $(n-2)$-uple dont chaque composante est dans $[n]$.

On peut vérifier que tout arbre sur $[n]$ peut être reconstruit à partir de son code de Prüfer, et que tout $(n-2)$-uple dont chaque composante est dans $[n]$ est le code de Prüfer d'un arbre sur $[n]$.

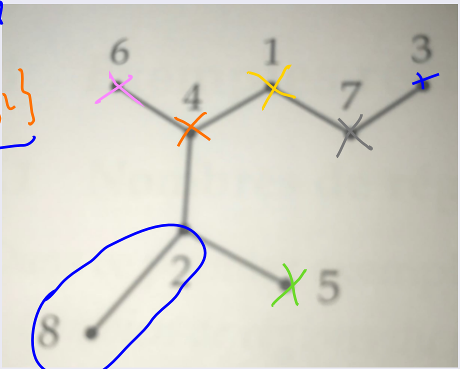
Comment trouver le code d'un arbre :

Exemple 35

Construire le code de Prüfer de l'arbre suivant.

$L = \{1, 2, 3, 4, 5, 6, 7, 8$

$S = \underbrace{\{7, 2, 4, 1, 4, 2\}}_{6\ (8-2)}$



Inversément, comment trouver l'arbre étiqueté d'après le code :

Exemple 36

Construire l'arbre associé au code de Prüfer suivant $(7, 2, 4, 1, 4, 2)$.

Comment procéder :

- Pour un code de longueur $n - 2$, lister les sommets $L = \{1, ..., n\} = [n]$
- Choisir le sommet étiqueté dans L le plus petit qui n'apparait pas dans le code S et le relier au premier sommet du code S.
- Retirer du code S le sommet utilisé pour créer S' et retirer le sommet étiqueté de la liste des sommets pour créer la liste L'.
- Arrêter lorsqu'il ne reste que 2 sommets dans la liste et plus rien dans le code, et relier ses deux sommets.

→ n'apparait pas dans S

$L = \{1, 2, 3, 4, 5, 6, 7, 8\}$

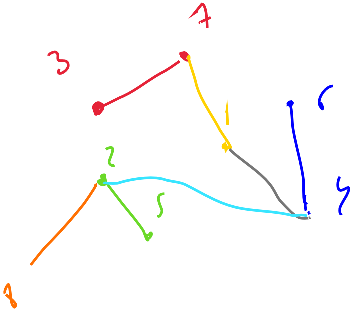
$S = \{7, 2, 4, 1, 4, 2\}$

$L' = \{1, 2, 3, 4, \cancel{5}, 6, 7, 8\}$

$S' = \{\cancel{7}, \cancel{2}, 4, 1, 4, 2\}$

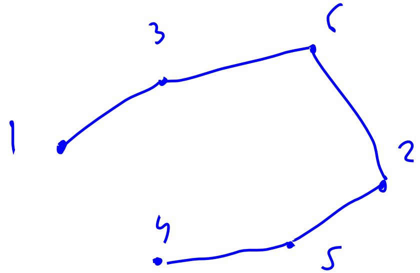
$L'' = \{\cancel{1}, 2, \cancel{4}, \cancel{6}, \cancel{7}, 8\}$

$S'' = \{\cancel{6}, \cancel{1}, \cancel{4}, \cancel{2}\}$



$S = (\cancel{3}, \cancel{4}, \cancel{8}, \cancel{2}) \to 6$ sommets

$L = \{\cancel{1}, 1, \cancel{3}, \cancel{4}, \cancel{8}, 6\}$



Pourquoi le code est-il forcément un $(n-2)$-tuple si l'arbre $T=(V,E)$ possède n sommets ? Notons d'abord plusieurs choses :

- Aucune feuille n'est présente dans le code ;
- Chaque sommet v est présent $\mid deg(v)-1 \mid$ fois dans le code ;
- Etant donné que T est un arbre, $|E|=n-1$.

Dans tout graphe, la somme des degrés est exactement deux fois le nombre d'arêtes (car chaque arête a deux extrémités). Donc en particulier pour notre arbre T : $\sum_{v\in V} deg(v)=2|E|$.

Le nombre de composantes du code est

$$\begin{aligned} \mid S \mid &= \sum_{v\in T}(deg(v)-1) = \sum_{v\in T} deg(v) - \sum_{v\in T} 1 = 2|E|-n \\ &= 2(n-1)-n = n-2 \end{aligned}$$

.

Théorème 37

Un vecteur d'entiers strictement positifs $(d_1, \ldots, d_n)$ *est le vecteur des degrés d'un arbre à* $n \geqslant 2$ *sommets (c'est-à-dire il existe un arbre à* n *sommets dont le* i*ème sommet a pour degré* d_i*) si et seulement si*

$$\sum_{i=1}^{n} d_i = 2n - 2 \ .$$

Coefficients multinomiaux : Application et Arbres

Démonstration Pour la parti "si" de la démonstration : Pour notre arbre $T = (V, E)$,

n sommets

$$\sum_{v \in V} deg(v) = 2|E| \ .$$

Etant donné que T est un arbre, $|E| = n - 1$ et on trouve

$$\sum_{v \in V} deg(v) = 2n - 2 \ .$$

La somme des degrés des sommets d'un arbre est deux fois son nombre de sommets, moins deux. Autrement dit, le vecteur des degrés d'un arbre à n sommets est un vecteur de n entiers strictement positifs dont la somme vaut $2n - 2$. De manière surprenante, ceci est une caractérisation des vecteurs des degrés des arbres !

Montrons à présent le "seulement si", par récurrence sur n. Le résultat étant clairement vrai pour $n = 2$, supposons que $n \geqslant 3$ et que le résultat est vrai pour moins de n sommets.
Par hypothèse, tous les d_i sont au moins 1 (étant des entiers strictement positifs). Mais ils ne peuvent pas être tous égaux à 1 car sinon $\sum d_i \leqslant n < 2n - 2$. Donc un des d_i est au moins 2. Mais on ne peut avoir $d_i \geqslant 2$ pour tout i car sinon $\sum d_i \geqslant 2n > 2n - 2$. Donc il existe un indice j tel que $d_i \geqslant 2$ et un indice k tel que $d_k = 1$. On peut supposer que $k = n$. Par l'hypothèse d'induction,

$$(d_1, \ldots, d_{j-1}, d_j - 1, d_{j+1}, \ldots, d_{n-1})$$

est le vecteur des degrés d'un arbre à $n - 1$ sommets, car sa somme est $2n - 4 = 2(n - 1) - 2$.

$\Leftarrow$ $P(n)$: $\sum_{i \in V} d_i = 2n-2 \Rightarrow \exists$ arbre $(d_1, \dots, d_n)$

$n = 2$


arbre $d_1 + d_2 = 2 = 4-2$

hyp : vrai $< n$

Thèse vrai n $\forall i \begin{cases} d_i \geq 1 \\ d_i \in \mathbb{Z} \end{cases}$

il existe certains $d_i > 1$ $\sum d_i \leq n < 2n-2$ ↯

$\exists i : d_i \geq 2$

Mais impossible $\forall i \geq 2 \rightarrow \sum d_i \geq 2n > 2n-2$ ↯

$\exists j : d_j \geq 2$

$\exists k : d_k = 1$

Donc il existe un arbre T' ayant $n-1$ sommets et $(d_1, \ldots, d_{j-1}, d_j - 1, d_{j+1}, \ldots, d_{n-1})$ comme vecteur de degrés. Rajoutons un $n$ème sommet à cet arbre, ainsi qu'une arête entre le $j$ème sommet et le $n$ème sommet. On obtient un arbre T ayant $(d_1, \ldots, d_{j-1}, (d_j - 1) + 1, d_{j+1}, \ldots, d_{n-1}, 1) = (d_1, \ldots, d_n)$ comme vecteur de degrés. Ce vecteur est donc bien le vecteur des degrés d'un arbre à n sommets. $\square$

Coefficients multinomiaux : Application et Arbres

Maintenant que nous comprenons très bien les vecteurs des degrés des arbres à n sommets, voyons comment compter le nombre d'arbres ayant un vecteur de degrés donné.

Théorème 38

Soit $(d_1, \ldots, d_n)$ un vecteur de $n \geqslant 2$ entiers strictement positifs sommant à $2n-2$. Le nombre d'arbres à n sommets ayant $(d_1, \ldots, d_n)$ comme vecteur de degrés est

$$\binom{n-2}{d_1 - 1, \ldots, d_n - 1}.$$

Coefficients multinomiaux : Application et Arbres

Démonstration : (Ne pas connaitre pour l'examen)
Dans notre preuve du Théorème de Cayley (Th 28 Chap 3), nous avons défini une bijection associant à toute fonction de $[n]$ dans $[n]$ un arbre étiqueté à n sommets dont deux sont spéciaux : $\bigcirc$ ("extrémité gauche") et $\square$ ("extrémité droite"). Nous allons réutiliser cette bijection pour montrer le résultat.

Etant donné que $\sum d_j = 2n - 2$ et que les d_j sont des entiers strictement positifs, $d_j = 1$ pour au moins *deux* indices j. (Ceci traduit le fait que tout arbre possède au moins deux feuilles.)

Coefficients multinomiaux : Application et Arbres

Supposons, sans perte de généralité, que $d_1 = d_n = 1$. Maintenant, considérons toutes les fonctions $f : [n] \to [n]$ telles que

(i) $f(1) = 1$ et $f(n) = n$;

(ii) pour $2 \leqslant j \leqslant n-1$, le nombre d'indices i tels que $f(i) = j$ est exactement $d_j - 1$ (de manière plus concise, $|f^{-1}(j)| = d_j - 1$).

Le nombre de telles fonctions f est le coefficient multinomial

$$\binom{n-2}{d_2-1,\ldots,d_{n-1}-1} = \binom{n-2}{0,d_2-1,\ldots,d_{n-1}-1,0}$$

$$= \binom{n-2}{d_1-1,\ldots,d_n-1}.$$

Coefficients multinomiaux : Application et Arbres

Pour chacune de ces fonctions f, l'ensemble $\mathcal{M} = \{i_1, \ldots, i_m\}$ des sommets du graphe dirigé $\vec{G}(f)$ apparaissant sur un cycle dirigé contient toujours 1 (comme élément minimum : $i_1 = 1$) et n (comme élément maximum : $i_m = n$), car $f(1) = 1$ et $f(n) = n$. Il résulte qu'on a toujours $\bigcirc = 1$ et $\square = n$.

On peut vérifier que l'arbre correspondant à toute fonction f satisfaisant (i) et (ii) a $(d_1, \ldots, d_n)$ comme vecteur de degrés. En outre, on peut vérifier qu'en inversant la correspondance, tout arbre sur $[n]$ ayant $(d_1, \ldots, d_n)$ comme vecteur de degrés correspond à une fonction f satisfaisant (i) et (ii). En conclusion, nous avons une bijection entre les arbres sur $[n]$ ayant $(d_1, \ldots, d_n)$ comme vecteur de degrés et les fonctions $f : [n] \to [n]$ satisfaisant (i) et (ii). $\square$

Remarque :
Pour une autre preuve du Théorème 38, observer que tout sommet j apparaît exactement $d_j - 1$ fois dans le code de Prüfer d'un arbre sur $[n]$ ayant $(d_1, \ldots, d_n)$ comme vecteur de degrés. Ceci donne l'idée de compter le nombre de mots de $n-2$ symboles pris dans l'alphabet $[n]$ où le symbole j apparaît exactement $d_j - 1$ fois. On sait par ce qui précède que ce nombre est $\binom{n-2}{d_1-1,\ldots,d_n-1}$. Ce nombre est bien le nombre d'arbres à n sommets ayant $(d_1, \ldots, d_n)$ comme vecteur de degrés (bien que correct, ceci demande à être justifié).

Résumé des points importants du chapitre

1. Arbres et les forêts ;
2. Lien entre les arbres et le chemin unique, un arbre est un graphe biparti ;
3. Les arbres enracinés : parent, enfant, frère, ancêtres, descendants, feuilles (sommets terminaux) et sommets internes, sous-arbres ;
4. Arbres *m*-aire, et *m*-aires entiers ;
5. Arbres enracinés ordonnés ;
6. Quelques applications en chimie et autre.... ;

Résumé des points importants du chapitre

1. Compter les sommets et les arêtes, les feuilles ;
2. Hauteur d'un arbre et niveaux dans un arbre, Arbres équilibrés ;
3. Arbres d'expressions ;
4. Arbres sous-tendants, et sous-tendants minimaux et algorithmes ;
5. Lien avec les énumérations : preuves bijections : Cayley, Polygones ;
6. Lien avec les énumérations : Coefficients multinomiaux et arbres : Code de Prüfer, Degrés.

Vivement la semaine prochaine pour de nouvelles aventures mathématiques !