Algorithmie : des maths et du computer science

- mathématicien persan du IXe siècle, Mohammed ibn-Musa al-Khuwarizmi
- logiciens des années 1930 (Herbrand, Gödel, Church et Turing)
- branche de l'informatique qui traite des algorithmes ou économie de l'informatique

Economie de linformatique – combine ca nous coute (le cout = le temp) dun algo

Coût d'un algorithme

- Expérimentalement :

  $time java Simulation 1000000

  $13.240u 0.870s 0:19.30 73.1% 0+853k 0+16io 0pf+0w

- Théoriquement :

  *Execution time = **instruction count (IC)** //nombre d'instructions exécutées du programme*

  x average clock per instruction (CPI) //nombre moyen de cycles par instruction

  x clock time (CT) // durée d'un cycle d'horloge

Les 2 cest Un nombre, cest un count

En seconde

- IC approximable par l'étude de la complexité
  - *Mesure de la complexité : 0(n) O(n^2) 0(log n) dépendant seulement du langage source*

Time – command linux pour demander le temp dexecution du pgm java « Simulation »

1. Temp user
2. Temp système
3. 3. Estimation du temp globale du code du moment qu'on la lancer jusc quil sarette.

Dans ce cas : 73.1% de la CPU pendant un certain temp
puit les entre-sortie des bus, autre architecture de la machine.

Mesur de complexiter – dans un monde ideal combien de temp lalgo prent, c une mesure moyenne, mais ca donne pas un nombre de seconde exact que ca va prendre, mais on a ne esstimation

Complexiter : esstimation globale du nombre dinstruction elememntaire, principale : parcourir un tableaux etc..

O(n) complexiter linaire
o(n^2) complexiter _____
on est pas content quand la complexiter est exponenssiele.

Algorithmie : résolution de problèmes via structures de données

- un entier n ; n est-il premier ? *L'algorithme stochastique de Miller-Rabin*
- un système d'équations linéaires ; le système est-il régulier ? si oui, calculer sa solution. *L'algorithme de triangulation de Gauss*
- un n- uplet de nombres complexes ; calculer la transformée de Fourier discrète de ce n-uplet. *La transformée de Fourier rapide*
- une chaîne de caractères ***s*** ; ***s*** est-elle un programme Java correct ? *L'analyse syntaxique par un automate à pile*
- un ensemble de villes et de routes les reliant, la longueur de ces routes, et deux villes de cet ensemble ; calculer la longueur du chemin le plus court entre ces deux villes. *L'algorithme du plus court chemin de Floyd, par programmation dynamique*
- un ensemble de tâches, leurs durées et leurs contraintes de précédence ; ces tâches sont-elles réalisables ? si oui, calculer leurs dates de réalisation au plus tôt et au plus tard. *Un algorithme glouton d'affectation de tâches*
- un réseau de transport de marchandises, la capacité de chaque route, et un entrepôt ; quelle est la quantité maximale de marchandises que ce réseau peut écouler à partir de l'entrepôt ? *L'algorithme de flot maximum de Ford-Fulkerson*
- comment rétablir le réseau électrique après une tempête en réparant le minimum de lignes électriques ? *L'algorithme glouton d'arbre couvrant minimum de Prim, ou celui de Kruskal*

1 - si on choisie une mauvaise structure de donnee on aura du mal a résoudre un pbm

La ya une liste de pbm, par exemple le premier c un algo pour savoir si n est premier

Algorithmie : résolution de problèmes via structures de données

Pas de livre magique mais des stratégies universelles :

- **la méthode incrémentale** : résoudre un problème P(n) à partir d'une solution de P(n-1) ; une méthode par récurrence, qui peut donner lieu aussi bien à des programmes itératifs qu'à des programmes récursifs ; dans le cas d'un problème d'optimisation, la méthode gloutonne consiste à construire une solution de P(n) en prolongeant une solution de P(n-1) par un choix localement optimal ;
- **la méthode << diviser pour régner >>** : méthode descendante par décomposition en sous-problèmes en transformant un problème P(n) en deux problèmes P(n/2) ;
- **la programmation dynamique** : méthode ascendante, utilisable pour des problèmes d'optimisation, qui consiste à construire la solution d'un problème à partir des solutions de tous les sous-problèmes ; elle s'applique quand toute sous-solution d'une solution optimale est optimale pour le sous-problème correspondant.

Mais le coeur du problème c'est la **Structure de Données (Data Structure)** : la modélisation du problème (monoïde, graphe, matroïde en math / pile, file, tas, table, arbre en info → math discrète)

Peut-on tout résoudre par un algorithme ? P=NP ? En un temps raisonnable ? Le fameux problème du Voyageur de Commerce

Strategie pour trouver quelle algo correspond pour résoudre un probleme.

Methode incrementale -> par recurrence.

.

Divider pour ranyer : trier un tableau dentier (algo vu l'année dernier)
on divise le pbm en deux -> ensuite on fusionne le resultat

Pas de livre magique pour savoir comment résoudre un pbm
mais

Etape 1 – choisire la bonne structure de donnee (le ceur du pbm) – déjà en penssan a ca la resolution du pbm est bcp plus simple.

Etape 2 – choisir la strategie

Les graphe – un objet abstrai qui se numerise très bien.

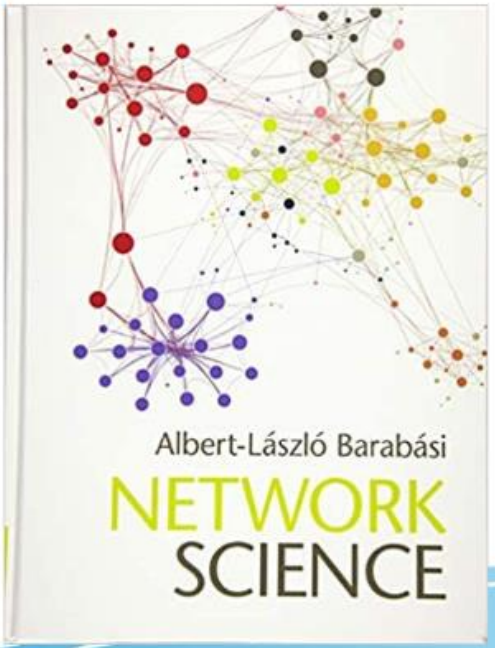
Algo avancee cest un peu les math discrete

Peut on résoudre tout les pbm par un algo ?

P = NP ? pbm ouvert en informatique, tjr pas resoulu

Être capable de comprendre et d'analyser un monde interconnecté
De la **théorie des graphes** vers la théorie des réseaux

(From Graph Theory to Network Science ; The Network Science by Prof. Albert-László Barabási)



Savoir passer des math pur a lanalise des réseaux
ce livre est dispo gratuit sur internet

Ya un chapitre sur la theorie des graphe

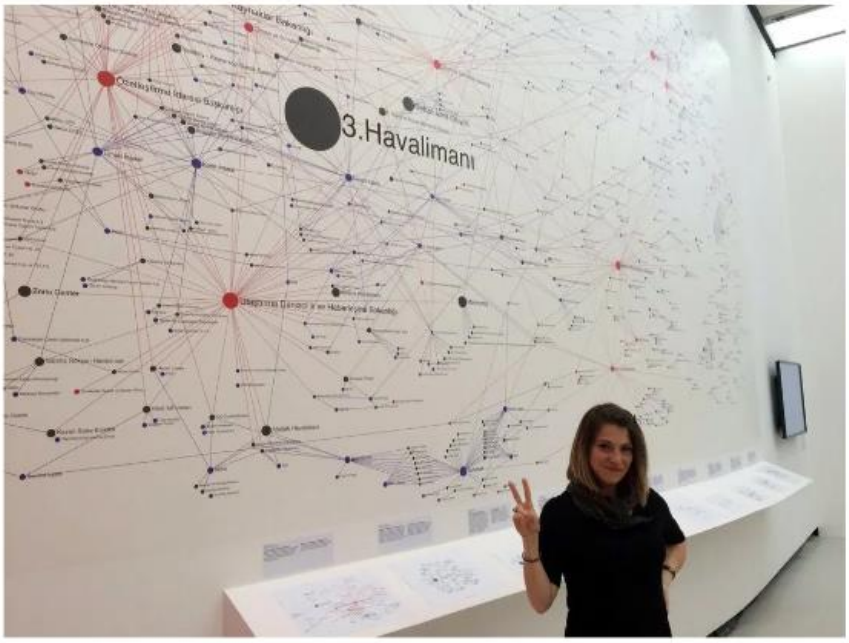
On peut sisnspirer de sertaine idder pour avancee

Le bute est pas de faire tout le livre.

Ya un magasin de livre pas loin de la fac avec une etage au sou saul avec bcp de livre math info.

Exploration via la théorie des Graphes/Réseaux

http://graphcommons.com



De la théorie des Graphes

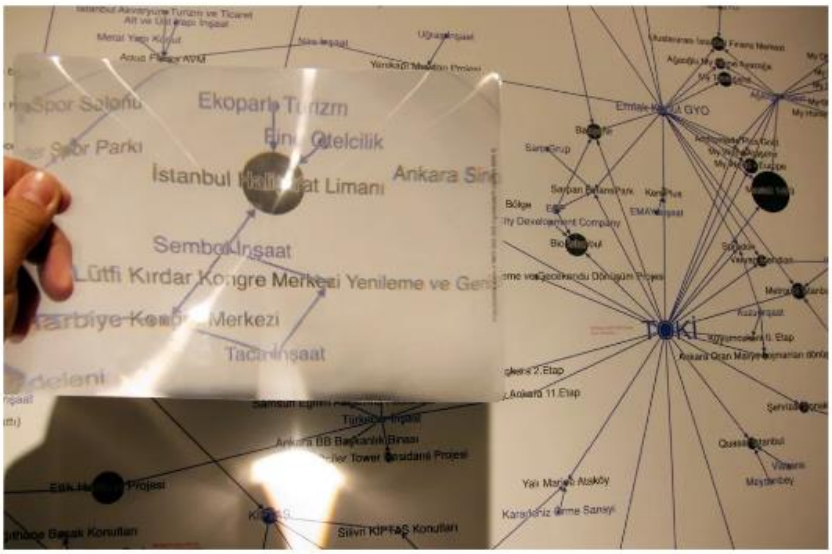
à la Science des réseaux

Etude des réseaux **réels** ;
Gephi , Cytoscape, NetworkX

On va explorer la theorie des graphes/réseaux ensemble. De poin de vu theorique et appliquer.

Sur le lien on peut voir des photo de ce genre

Exploration via la théorie des Graphes/Réseaux



Pbm ergonomique – comment me balader dans un graph qui a 1000 neu de façon utile, agréable

Ya une difference entre la theorie des graphe (proprieter abstrai) et tt les outils qui exisste pour les utiliser vrmt

Il y a qulque outils :

Etude des réseaux **réels** ;
Gephi , Cytoscape, NetworkX

Phython c pas mal, ya des module pour visualiser des graph
et ya des logiciel gratuit comme Gephi

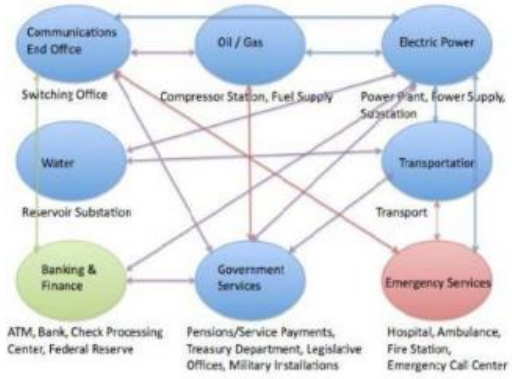
Cytoscape – plus utilizer par les biologistes
NetworkX – le module python le plus simple pour manipuler des graphe.

3 outils qu'on peut déjà installer sur lordi pour samuser.
Outils dispo gratuit

Qulque ilustration encore :

Exploration via la théorie des Graphes/Réseaux

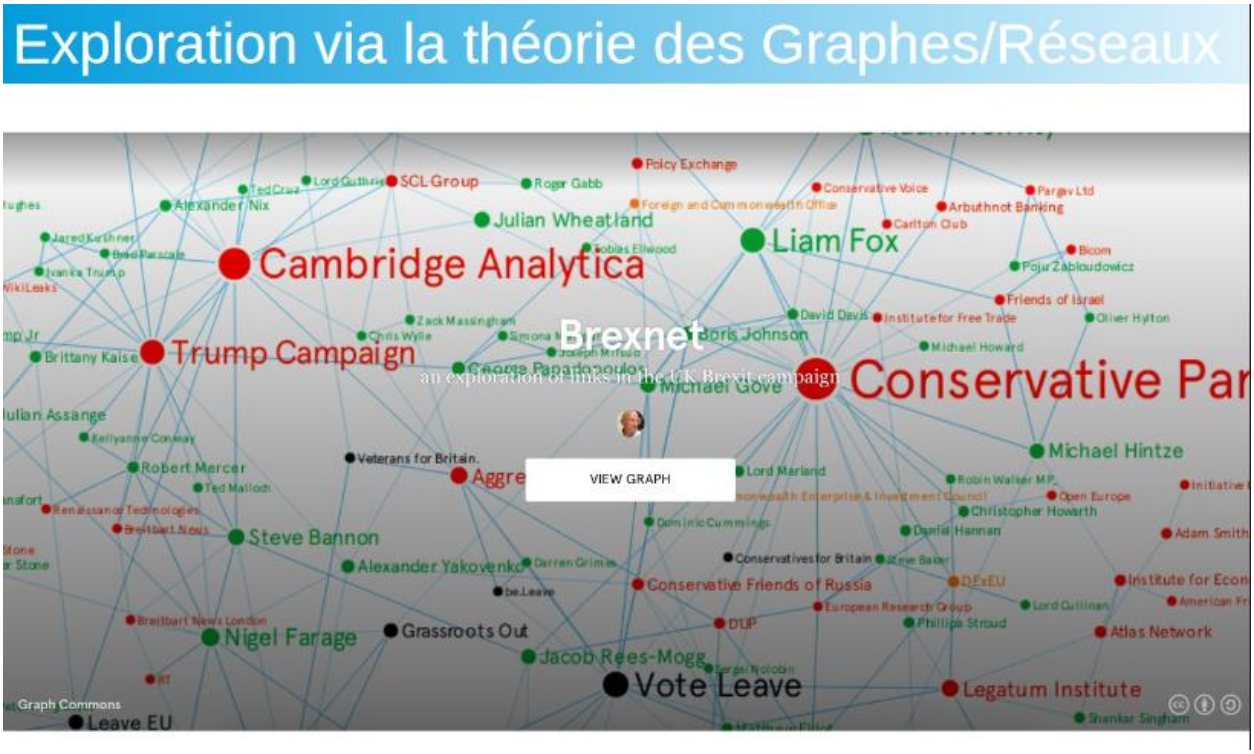
Figure 5: Essential Services Interconnectedness Affected by Power Grid Outage



Changement climatique : un rapport commandé par le Pentagone alerte sur les risques à venir

Si lalimentation pétrole est stopee -> quessque va se passer ?

Encore une exemple :

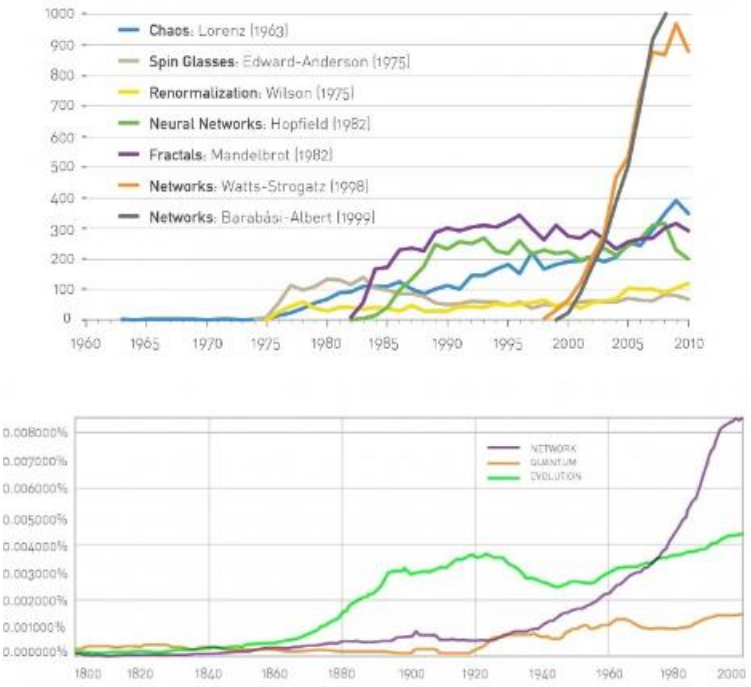


Façon dexplorer la campagne du brexit en Britanie
Ca capelle Graph Commun
https://graphcommons.com/
https://graphcommons.com/graphs/a936d1c7-2b52-47ba-bc21-79a2631149d0?auto=true

on peut sinscrire sur ce site pour samuser un petit peu

1

Exploration via la théorie des Graphes/Réseaux



Science des Réseaux	Theorie des Graphes
Réseau *Network*	Graphe *Graph*
Noeud *Node*	Sommet *Vertex*
Lien *Link*	Arête *Edge*

Exploration devalutaion de mot clé dans la literature… une autre manière dutiliser les graphes.

Si on fait des math pur on va parler dobjets comme les sommets (vertex en anglais)…. Etc
si on parle de reseaux on parle d'instatentation dans le monde reel/physique, on parlera de lien plutôt que darete mais ca veut dire la meme chose.

CETTE annee on va commencer de comprendre la theorie des graphes de poit de vu mathematique et informatique appliquee.

http://visualcomplexity.com/vc/

des graphe complex -> comment visualizer ?
liee a la complexité
par exemple si on tappe sur biologie on peut voir les relation dinteraction entre les proteinne (cest aussi la theorie des graph), util pour letude de maladies

(diapos cours de l'année dernier) – jusc au diapo 13 inclus.

Théorie des Graphes 1

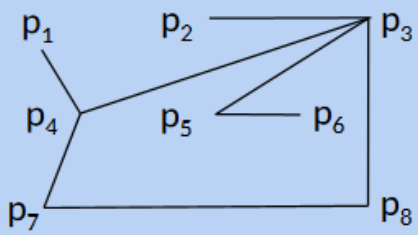
Problème classique de coloration de graphes :
On doit stocker 8 produits chimiques p 1 , p 2 ... p 8 mais pour des raisons de sécurité certains produits ne peuvent pas être stockés dans le même hangar :

- p 1 ne peut pas être stocké avec p 4
- p 2 ne peut pas être stocké avec p 3
- p 3 ne peut pas être stocké avec p 2, , p 4 ou p 5
- p 4 ne peut pas être stocké avec p 1, , p 3 ou p 7
- p 5 ne peut pas être stocké avec p 3 ou p 6
- p 6 ne peut pas être stocké avec p 5
- p 7 ne peut pas être stocké avec p 4 ou p 8
- p 8 ne peut pas être stocké avec p 3 ou p 7

Combien de hangars faut-il et comment y répartir les produits ?

Résolution :
1. Représenter le problème sous-forme de graphe.
Sommets = produits
Relation = « n'est pas compatible avec »
(Deux produits sont adjacents s'ils ne peuvent pas être stockés dans le même hangar.)
La relation est symétrique : graphe non orienté

2. Attribuer un hangar à chaque sommet de telle sorte que 2 sommets reliés par une arête n'est pas le même hangar.
C'est un problème de **coloration du graphe**.



Modéliser le pbm – trouver la structure de donneess.

Théorie des Graphes 1

Étant donné un graphe simple G = (S,A), S ensemble des sommets, A ensemble des arêtes.
Une coloration propre des sommets est une fonction f qui attribue une couleur (valeur) à chaque sommet de sorte que deux sommets adjacents aient des couleurs différentes : $\forall (x, y) \in A, f(x) \neq f(y)$

• Un graphe est k-colorable s' il existe une coloration propre avec k couleurs

• Le nombre minimum de couleurs nécessaires pour obtenir une coloration propre est appelé le nombre chromatique de G et noté $\chi(G)$. C'est le plus petit entier k pour lequel le graphe est k-coloriable. Ki de G

Théorie des Graphes 1

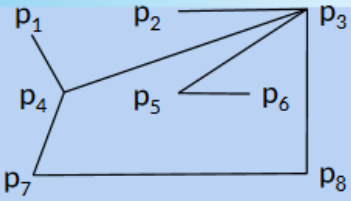
Problématiques :
1. Etant donné un graphe simple G d'ordre n, comment attribuer une « couleur » (valeur) à chaque sommet de G de telle sorte que deux sommets adjacents n'aient jamais la même couleur ?
2. Etant donné un entier k<n, G est-il k-colorable ?
3. Quel est le nombre chromatique de G ?
4. Peut-on trouver une coloration minimale ?

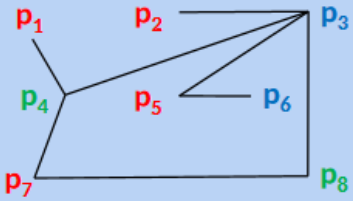
Domaines d'applications :
Cartographie
Planification (réunions, emploi du temps ...)
Transports, stockages (problèmes d'incompatibilité)
Télécommunications : attribution de fréquences
Informatique : allocation de registres

Théorie des Graphes 1

A. Algorithme naïf : coloration séquentielle

− Parcourir les sommets les uns après les autres :
− Attribuer à chaque sommet la « plus petite » couleur non déjà utilisée pour un de ses sommets adjacents.





sommets	couleurs
1	1
2	1
3	2
4	3
5	1
6	2
7	1
8	3

Si n et m sont respectivement l'ordre (nombre de sommets) et la taille (nombre d'arêtes) du graphe, la complexité de l'algorithme séquentielle est en O(n+m)
C'est un algorithme glouton. La solution obtenue n'est pas forcément optimale et dépend de la numérotation des sommets (de l'ordre de Parcours)

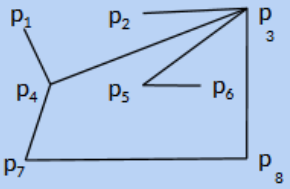
On peut améliorer l'algorithme séquentiel en choisissant un ordre de parcours judicieux → Algorithme de Welsh et Powell

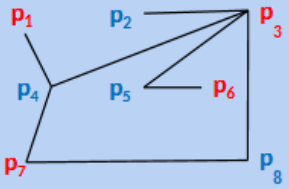
Algo glouton – on obtien la solution petit a petit…..

Théorie des Graphes 1

B. Algorithme de Welsh et Powell

1. Trier la liste des sommets par ordre décroissant de degrés
2. Parcourir la liste triée :
   − Trouver le premier sommet non déjà coloré et lui attribuer la « plus petite » couleur c non déjà utilisée.
   − Parcourir la suite de la liste en attribuant cette couleur c aux sommets non colorés et non adjacents aux sommets déjà colorés avec c
3. Recommencer en 2 s'il reste des sommets non colorés





sommets	degre	t 1	t 2
3	4	• 1	
4	3		• 2
5	2		• 2
7	2	• 1	
8	2		• 2
1	1	• 1	
2	1		• 2
6	1	• 1	

Quel est l'idée derrière cette stratégie ?
Obtient-on la bonne solution ?
L'algorithme donne-t-il toujours le nombre chromatique ?

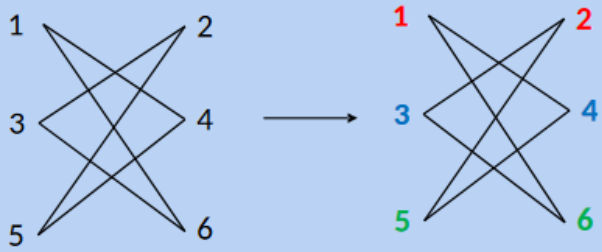
Lidder de base : si je commence a colorier les sommet avec le plus grand connecter : je commence avec ceu qui sont le plus dur a colorier et ensuite je colorie ceu qui sont plus simple a colorier.

Nb chromatique – ne donne pas tjr ce combre..

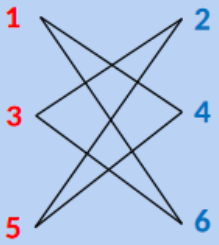
Théorie des Graphes 1

En utilisant un tri par dénombrement, on obtient une complexité de l'ordre de O(n+m). C'est un algorithme glouton. La solution obtenue n'est pas forcément optimale:

Graphe couronne à 6 sommets



L'algorithme donne 3 couleurs alors que le nombre chromatique est 2



Théorie des Graphes 1

On sait trouver une coloration d'un graphe mais **on ne sait pas résoudre en temps raisonnable les deux questions suivantes :**

- **Déterminer si un graphe est k-colorable**
- **trouver le nombre chromatique d'un graphe quelconque**

On apelle ca

Les algorithmes ont une complexité exponentielle (NP-complet et NP-difficile)

On sait encadrer le nombre chromatique :

− Le nombre chromatique d'un graphe G est inférieur à Δ(G), où Δ(G) est le plus grand degré de ses sommets.

− Le nombre chromatique d'un graphe G est supérieur ou égal à l'ordre ω(G) du plus grand sous-graphe complet (ou clique) de G.

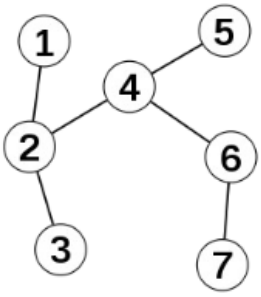
$$\omega(G) \leq \chi(G) \leq 1 + \Delta(G)$$

On sait trouver des colorations optimales pour certaines classes particulières de graphes (graphes parfaits par exemple)

Une structure de données simple

```
struct nœud {
    struct *noeud_filsG, *noeud_filsD ;
    int valeur ; // Ou tout autre structure
}
```

Un arbre vs nœud/sommet ? Nœud vs racine / Arbre binaire vs. Arbre générique



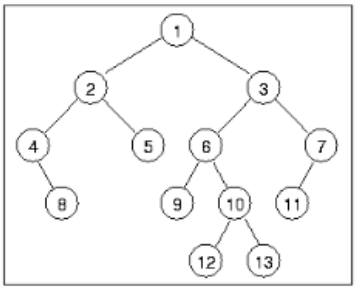


FIG. 2.14 – *Arbre binaire étiqueté.*

https://fr.wikipedia.org/wiki/Arbre_(th%C3%A9orie_des_graphes)

http://cermics.enpc.fr/polys/oap/node72.html

http://emmanuel.adam.free.fr/site/spip.php?article128

Cree des arbres binaires

Contrôle Continu Intégré

Note de session 1 :

Note 1= 0.2*Note A +0.3*Note B + 0.5*Note C

Note A : 1 compte rendu de Tps (à définir)

Note B : 1 examen intermédiaire si présentiel possible sinon mini projet à rendre

Note C : 1 examen final en présentiel obligatoirement

Note de Session 2 :

Note 2= max(Note C, Note 1)

Les coeff peuve encore changer

Pas de session2 (car rattrapge intgrer)

TPS – s car peut être ca va retre sur plusieurs TP, c pas encore dessider.

Sylvan – grp lundi
ALLAn – les grp de mardi