Méthodes d'optimisation

Arbre

Définition : Les arbres sont des graphes :

1) Sans cycle
2) Connexe

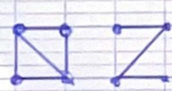
Sommet de deg 1 sont les feuilles et les autres sont des noeud.

Un arbre recouvrant d'un graphe G est un sous-graphe de G qui est un arbre.

Propriété :

Un graphe est connexe ssi il possède un arbre recouvrant.

Exemple.



Théorème de caractéristique des arbres

pour tous graphes ~~(X, Y)~~ : $G = (X, E)$

1) G est un arbre
2) Pour tous couple (x, y) de sommets distincts de G, il existe 1 unique chaine P_{xy} d'extrémité x et y.
3) G connexe
4) G sans cycle
5) G est connexe et possède $n - 1$ arrêtes
6) G est sans-cycle est possède $n - 1$ arrêtes.

Définition

> Un graphe $G = (X, E)$ est dit valué aux arrêtes s'il existe une application v de E dans IR $v(x, y)$ est la valuation xy.

> Valuation d'un chaine P, $v(P)$ = sommes des valuations de chaques arretes.

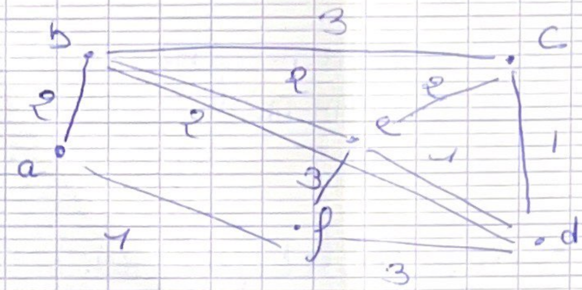
> Distance entre 2 sommets x et y, $d_{G,v}(x, y)$ = valuation minimum d'une chaine du graphe valué ayant ces 2 sommets pour extrémités.

Algorithme de Kruskal

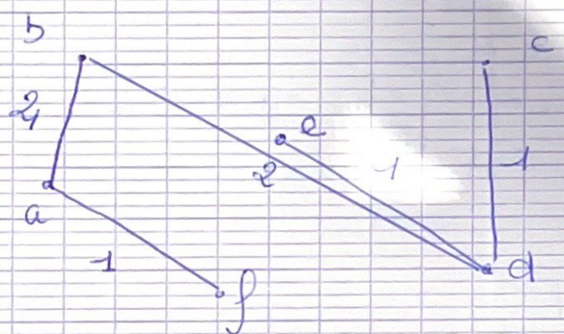
> ~~Classer~~ Retourne un arbre recouvrant de G de poids minimal.

```
DEBUT
| Classer les arrêtes de E par ordre croissant de
|         valuation. On note (e1, e2, ... ex) cet ordre
|         sur les arretes.
| H <- (X, Ø)
| Pour i de 1 à m faire
|       si H + ei est sans cycle alors
|             H <- H + ei
|       Fin Si
| Fin Pour
| Retourne H
FIN
```

Exemple:



T = (af, ed, dc, ab, bd, be, ec, bc, fd, fe)



Le poids de l'arbre courant minimal est 7

Algorithme de Prim-Jarnick

DEBUT

Choisir un sommet x de G

$S \leftarrow \{x\}$

$U \leftarrow \emptyset$

Tant que $A_G(S) \neq \emptyset$ alors

choisir une arrête e de $A_G(S)$ de valuation min.

noter y l'extrémité de e qui n'est pas dans S

$S \leftarrow S \cup \{y\}$

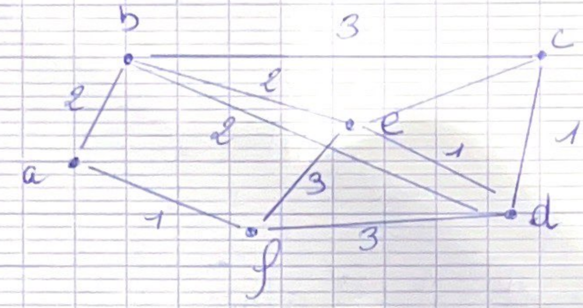
$U \leftarrow U \cup \{e\}$

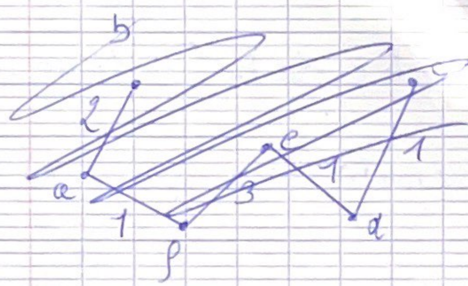
Fin Tant Que

renvoyer le graphe $A = (S, U)$

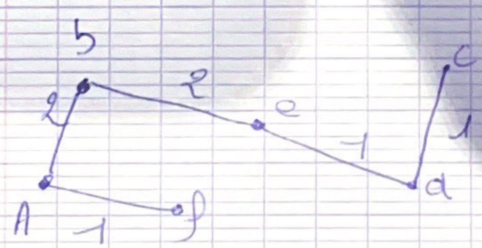
FIN

Exemple





~~étapes~~	~~S~~	~~U~~	~~$A_G(S)$~~
~~1~~	~~b~~		~~ba, bc~~
~~2~~	~~b, a~~	~~ba~~	~~a, f~~
~~3~~	~~b, a, f~~	~~ba, af~~	~~fe, fd~~
~~4~~	~~b, a, f, e~~	~~ba, af, fe~~	



étapes	S	U	$A_c(S)$
1	b		ba, bc
2	b, a	ba	be, af
3	b, a, f	ba, af	be, fe, fd
4	b, a, f, e	ba, af, be	ea, fd, bc
5	b, a, f, e	ba, af, be, ed	
6	b, a, f, e, d	ba, af, be, ed, dc	