

I. Questions de cours

1. Arbre couvrant de poids minimum. Sur $G = (V, E)$

A est un graphe connexe sans cycle $A=(V_A, E_A)$

$V = V_A$

E_A inclus dans E.

Soit \mathcal{A} l'ensemble des A inclus dans G. On appelle un ACPM un arbre a.

Tel que : $\sum_{e \in E_A} w_e = \min_{e \in A} \sum_{e' \in E_B} w_{e'}$

2. Démonstration unique ACPM

Cf. la démonstration présente dans un autre document.

II. Déroulement d'un algorithme 1

i	C	bip	bop	truc	(q)	Premier tour	Deuxième tour
1	1	X 8	X	Ev arg	10		
5	5 1	X 1 4	X	Ev arg	8 3		
2	2	X 5	X	Ev arg	1		
7	7 2 1	X 2 1	X	ev	8 4		
6	6	X 7	X	Ev arg	2		
3	3	X 6	X	Ev arg	7		
11	11 3	X 3 2	X	Ev arg	6 1		
9	9	X 11	X	Ev arg	3		
4	4	X 9	X	Ev arg	11		
10	10 4	X 4 3	X	Ev arg	9 2		
8	8	X 10	X	Ev arg	4		

Meilleurs cas pour 1 initiateur on a $2n$, et pour n initiateurs on a $4n$.

Pires cas pour 1 initiateur $2n$, et pour n initiateurs $2n((\log_2 n) + 1)$.

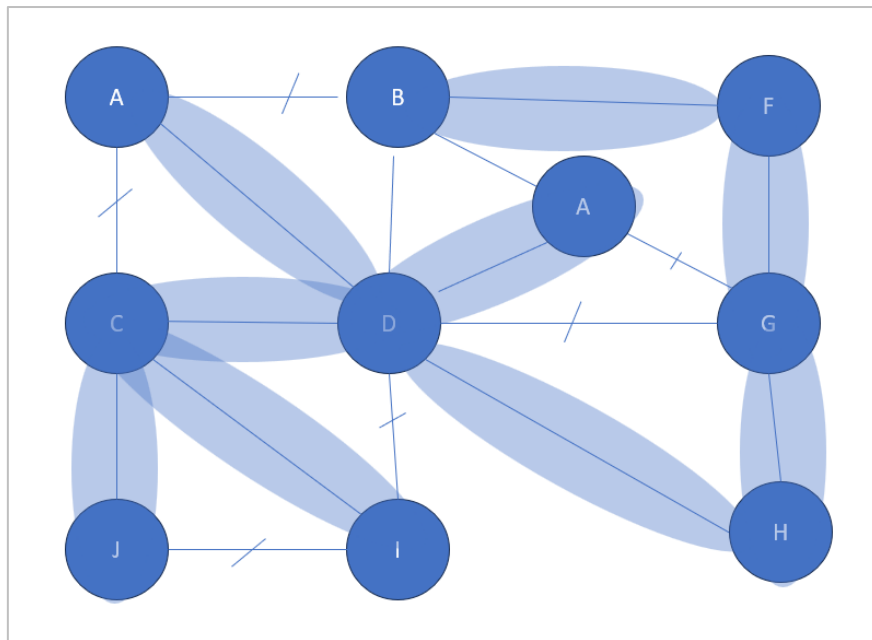
III. Déroulement d'un algorithme 2

1. Expliquer le principe du fonctionnement

Maintient un ensemble de fragments avec deux règles : absorbions et fusions. Si les niveaux des fragments sont différents celui avec le plus grand niveau absorbe le plus petit. S'ils sont équivalents ils fusionnent.

2. Expliquer le déroulement

Revoir sur le schéma de l'énoncé les valeurs des arrêtes. Penser à expliquer son cheminement. Voici le rendu final :



3. Dérouler l'algorithme

	A			B		C		D		
	B	C	D	A	D	A	D	A	B	C
Card	B Reject	B Reject	Br	B Reject	Br	B Reject	Br	Br	B Br	B Br
Niv	0 1			0 1		0 1		0 1		
Etat	Found Find Found			Found Find Found		Found Find Found		Found Find Found		
Recu	0			0		0		0 1 2		
Nom	3			3		3		3		
Père	D			D		D		A		
Mcan	X			X		X		X		
Testcan	C X			A X		A X		X		
Mpoids	Infini			Infini		Infini		Infini		

Messages à traiter :

- A → D connect 0
- D → A connect 0
- B → D connect 0 *traiter plus tard*
- C → D connect 0 *traiter plus tard*
- A → D initiate 1,3,find
- D → A initiate 1,3,find
- A : test
- D : test
- D → B initiate 1, 3, find
- D → C initiate 1, 3, find

- B : test
- C : test
- A → C test 1,3
- B → A test 1,3
- C → A test 1,3
- D report *se passe rien*
- C test
- A → B reject
- A test
- B test
- A report
- B report
- C report
- A → D report infini **terminé**
- B → D report infini *fait incrémenter le reçu et exécute la procédure report*
- C → D report infini *fait incrémenter le reçu et exécute la procédure report*
- D → A report infini **terminé**

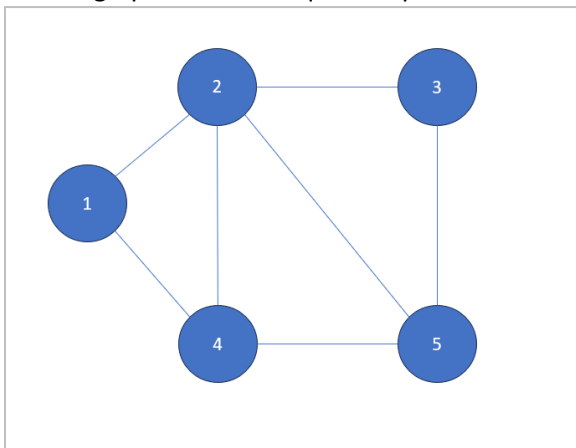
IV. Déroulement d'un algorithme 3

1. Rappeler les hypothèses nécessaires à cet algorithme

- Graphe valué
- Poids des arrêtes potentiellement négatifs mais pas de cycles négatifs

2. Dérouler cet algorithme sur un graphe valué de 5 sommets de votre choix

Voici le graphe en disant que chaque arrête vaut 1 :



Initialisation :

	1	2	3	4	5
1	1	0	INFINI	1	INFINI
2	1	0	1	1	1
3	INFINI	1	0	INFINI	1
4	1	1	INFINI	0	1
5	INFINI	1	1	1	0

K= 0 :

	1	2	3	4	5
1	1	0	INFINI	1	INFINI
2	1	0	1	1	1
3	INFINI	1	0	INFINI	1
4	1	1	INFINI	0	1
5	INFINI	1	1	1	0

K= 1 : on utilise le nœud intermédiaire 1

	1	2	3	4	5
1	1	0	2	1	2
2	1	0	1	1	1
3	2	1	0	2	1
4	1	1	2	0	1
5	2	1	1	1	0