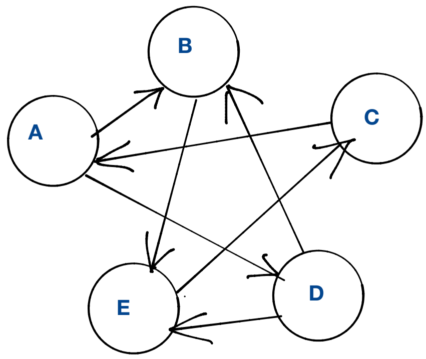
Correction du médian P2020

ENONCE

Exercice 1 (3 points)
Soit G, le graphe donné ci-dessous. Attention, chaque mauvaise réponse est sanctionnée par -0,5 points.
Répondre aux questions suivantes (vrai ou faux)
1. Le graphe G est planaire.
2. Le graphe G admet une bonne numérotation.
3. Le graphe G admet un chemin Eulérien.
4. Le graphe G admet un circuit Hamiltonien.
5. Le graphe G est biparti.
6. Le nombre chromatique du graphe G est 3.



Graphe G

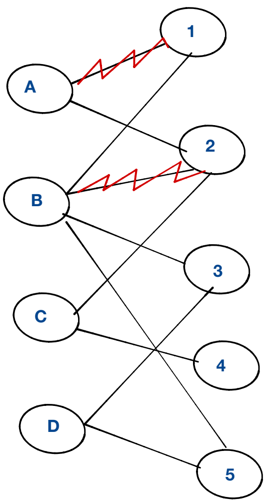
Exercice 2. (3 points)

Appliquer l'algorithme de couplage maximal (donné dans le polycopié) sur le graphe biparti G ci-dessous à partir d'un couplage initial {(A,1),(B,2)}. A chaque itération, le marquage initial (*) des sommets dans X se fera dans l'ordre A, B, C, D, et l'examen des sommets marqués se fera sur l'ordre FIFO (premier marqué, premier examiné).

Q1. Quelle est l'arête rajoutée au couplage à la fin de la 1ère itération?
1 choix multiple :

- **(B,3)**
- **(C,3)**
- **(C,4)**
- **(D,4)**
- **(D,5)**

Q2. Quel est le cardinal du couplage maximal ?
Q3. Parmi les ensembles suivants, quels sont ceux qui sont des couplages maximaux ? Attention, chaque mauvaise réponse est sanctionnée par -0,25 points.
{(A,2), (B,1), (C,4), (D,3)}
{(A,1), (B,3), (C,4), (D,5)}
{(A,1), (B,3), (C,4), (D,3)}
{(A,1), (B,2), (C,4), (D,5), (B,3)}
{(A,2), (B,5), (C,4), (D,3)}
{(A,1), (B,3), (C,2)}
{(A,2), (B,1), (C,4) (D,4)}



Graphe G

Exercice 3. (3 points)

On rappelle qu'un arbre est un graphe connexe et sans cycle. On a étudié dans le TD4 les liens qu'il y a entre graphe biparti et graphe 2-colorable.

Question 1.
Servez-vous des propriétés démontrées lors du TD4 pour prouver que tout arbre est un graphe biparti. On considère ici le cas des graphes non orientés. Justifier votre réponse.

Algorithme de calcul d'un arbre couvrant de coût minimum. (11 points)
Pour un graphe donné, un arbre couvrant est un graphe partiel connexe et sans cycles. En d'autres termes, l'arbre couvrant contient (couvre) tous les sommets du graphe et les arcs de l'arbre connectent tous les sommets. Ainsi, pour un graphe de n sommets, un arbre couvrant contient n-1 arêtes et tous ses sommets. Soit un graphe non orienté connexe G(V,E,w) de n sommets et m arêtes où w est une valuation des arêtes. On suppose que tous les arêtes sont de poids différents et strictement positifs.

Algorithme Couv(G)
Début
1. Lire le graphe G(V,E,w)
2. S = {a};
3. T = {}; //T est l'ensemble des arêtes de l'arbre initialement vide//
4. tant que (|S| < |V|)
5. (u,v) = arête de poids min ayant $u \in S$ et $v \in V-S$;
6. S = S$\cup${v};
7. T = T $\cup${(u,v)};
8. Fin tant que
9. Fin

//Explication : On part d'un sous-graphe partiel réduit à un seul sommet du graphe. À chaque itération, on agrandit le sous-graphe partiel en lui ajoutant le sommet libre accessible relié par l'arc du plus petit poids. On s'arrête quand tous les sommets sont joints à l'arbre.//

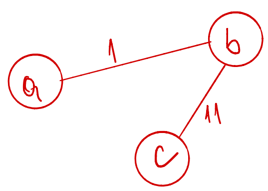
Pour implémenter cet algorithme, S est géré avec un tableau de booléens où S[i]=vrai si i appartient à S et faux sinon. T est géré avec un tableau d'arêtes initialement vide où l'on ajoute les arêtes l'une derrière les autres. Pour le codage du graphe on peut utiliser soit la matrice associée, soit les files des successeurs. Le graphe étant non orienté, une arête (i,j) est représentée par les deux arcs (i,j) et (j,i) dans les structures de données. Ainsi, la matrice associée obtenue est symétrique. Les poids sont stockés avec le codage du graphe, p.ex. dans la matrice associée, l'element (i,j) donne le poids de l'arête (i,j), tandis que pour les files des successeurs on rajoutera une file beta' similaire à beta mais qui contient le poids de l'arc associé, ainsi p.ex. le sommet correspondant au premier successeur du noeud i sera donné par beta(alpha(i)) et le poids de l'arc (i, beta(alpha(i))) est stocké dans beta'(alpha(i)).

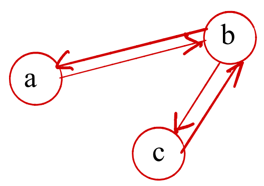
Question 1. Appliquer l'algorithme Couv sur le graphe G1. Rapportez les arêtes appartenant à la solution donnée par l'algorithme. (2 points)

Question 2. Donner la complexité de l'algorithme COUV dans le cas où le graphe est codé a) par la matrice associée b) par les files des successeurs. Détailler la complexité de l'algorithme pour chaque ligne et déduire la complexité globale pour chaque cas. Justifier. (3 points)

Question 3. Démontrer que l'algorithme se termine et qu'on obtient bien un arbre couvrant à la fin de l'algorithme. (3 points)

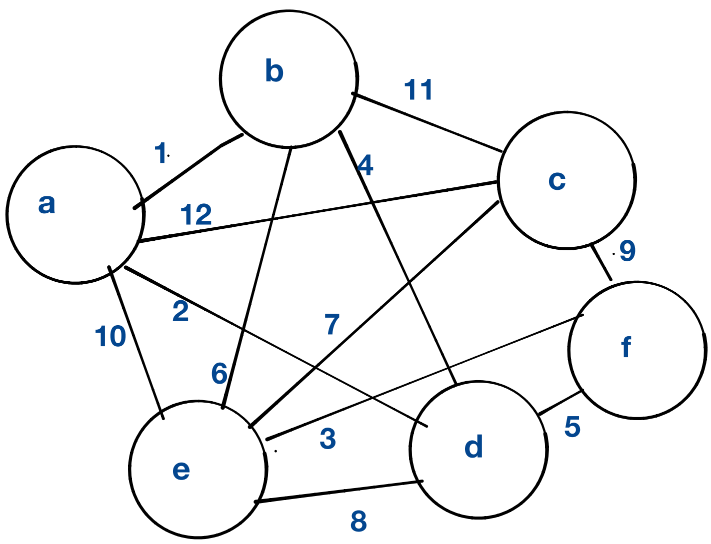
Question 4. Démontrer que l'arbre obtenu est de coût minimum. (3 points).





	a	b	c
a	0	1	0
b	1	0	11
c	0	11	0

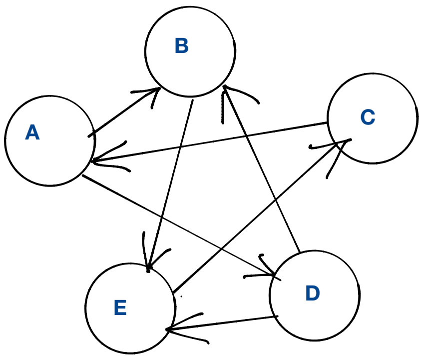
α	1	2	4	5
β	b	a	c	b
β'	1	1	11	11



Graphe G1

CORRECTION

Exercice 1 facile (3 points)
Soit G, le graphe donné ci-dessous. Attention, chaque mauvaise réponse est sanctionnée par -0,5 points.
Répondre aux questions suivantes (vrai ou faux)
1. Le graphe G est planaire. VRAI
2. Le graphe G admet une bonne numérotation.
3. Le graphe G admet un chemin Eulérien.
4. Le graphe G admet un circuit Hamiltonien. VRAI
5. Le graphe G est biparti.
6. Le nombre chromatique du graphe G est 3. VRAI



Graphe G

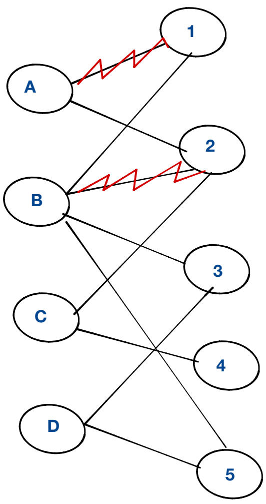
Exercice 2. (3 points)

Appliquer l'algorithme de couplage maximal (donné dans le polycopié) sur le graphe biparti G ci-dessous à partir d'un couplage initial {(A,1),(B,2)}. A chaque itération, le marquage initial (*) des sommets dans X se fera dans l'ordre A, B, C, D, et l'examen des sommets marqués se fera sur l'ordre FIFO (premier marqué, premier examiné).

Q1. Quelle est l'arête rajoutée au couplage à la fin de la 1ère itération?
1 choix multiple :

- **(B,3)**
- **(C,3)**
- **(C,4) VRAI**
- **(D,4)**
- **(D,5)**

Q2. Quel est le cardinal du couplage maximal ? (4)
Q3. Parmi les ensembles suivants, quels sont ceux qui sont des couplages maximaux ? Attention, chaque mauvaise réponse est sanctionnée par -0,25 points.
{(A,2), (B,1), (C,4), (D,3)} VRAI
{(A,1), (B,3), (C,4), (D,5)} VRAI
{(A,1), (B,3), (C,4), (D,3)}
{(A,1), (B,2), (C,4), (D,5), (B,3)}
{(A,2), (B,5), (C,4), (D,3)} VRAI
{(A,1), (B,3), (C,2)}
{(A,2), (B,1), (C,4) (D,4)}



Graphe G

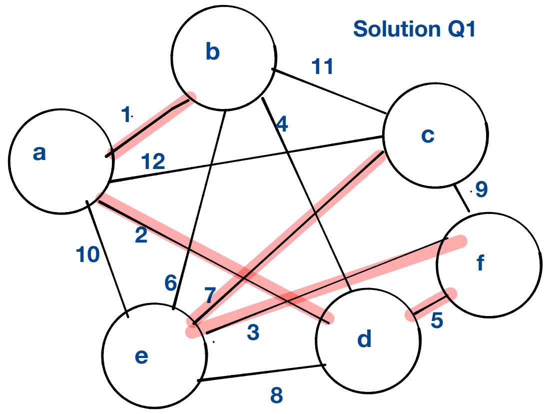
Exercice 3. (3 points)

On rappelle qu'un arbre est un graphe connexe et sans cycle. On a étudié dans le TD4 les liens qu'il y a entre graphe biparti et graphe 2-colorable.

Question 1.
Servez-vous des propriétés démontrées lors du TD4 pour prouver que tout arbre est un graphe biparti. On considère ici le cas des graphes non orientés. Justifier votre réponse.

La réponse est tres facile si on utilise la propriété que un arbre ne contient pas de cycles, donc ne contient pas de cycles de longueur impaire, donc est forcement biparti.

Algorithme de calcul d'un arbre couvrant de coût minimum.
Algorithme Couv(G)
Début
1. Lire le graphe G(V,E,w)
2. S = {a};
3. T = {}; //T est l'ensemble des arêtes de l'arbre initialement vide//
4. tant que (|S| < |V|)
5. (u,v) = arête de poids min ayant u∈S et v∈V-S;
6. S = S∪{v};
7. T = T ∪{(u,v)};
8. Fin tant que
9. Fin
//



Graphe G1

Q1. Ce sont les aretes :(ab), (ad), (df), (ef), (ec);

Question 2. Donner la complexité de l'algorithme COUV dans le cas où le graphe est codé a) par la matrice associée b) par les files des successeurs. Détailler la complexité de l'algorithme pour chaque ligne et déduire la complexité globale pour chaque cas. Justifier. (3 points)

Réponse: dans le cas où on utilise la matrice associée, le calcul de poids min (ligne 5) revient à O(n^2), tandis que dans le cas des alpha, beta, cela revient à O(m). Donc, complexité globale serait O(n^3) et O(n*m) respectivement.

Question 3. Démontrer que l'algorithme se termine et qu'on obtient bien un arbre couvrant à la fin de l'algorithme. (3 points)

Réponse: à la fin de l'algorithme on a inclu tous les sommets dans l'ensemble S, donc c'est bien connexe car à chaque itération on connecte un nouveau sommet avec le reste des noeuds par une arête et on rajoutera que n-1 arêtes;

Question 4. Démontrer que l'arbre obtenu est de coût minimum. (3 points).

Soit T l'arbre obtenu par l'algorithme et T' l'arbre de cout min réel. Notons e1, e2, ... en-1 les aretes ordonnées selon l'ordre d'inclusion dans l'arbre selon l'algorithme et S l'ensemble des sommets couverts par ces arêtes. Soit er la premiere arête de T différente de T', donc ne faisant pas partie de T'. En rajouteant er à T' on forme un cycle et il y a es qui fait partie de T' et qui joigne S du reste. On en conclut alors que T'' = T' -es + er est de cout < a T' et il est aussi un arbre couvrant. Absurde.