Partiel 2019

I. Chemins optimums (7 points)

On considère le graphe G suivant :

sommets i	a	b	c	d	e
successeurs j	b,c,e	c,e	d,e	a,c,e	/
poids des arcs (i,j)	5,3,10	4,2	2,6	4, 1, 4	/

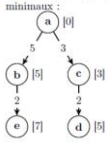
1. Calculez les chemins a-i-minimaux depuis le sommet a vers tous les autres sommets i (précisez l'algorithme utilisé, donnez son déroulement dans un tableau, et l'arborescence des chemins a-i-minimaux).

Corrigé

(2.5 pts) Les poids sont positifs, il y a au moins un circuit (cd) (dc), donc Dijkstra (et pas Bellman sans circuit) sera l'algo le plus efficace.

sommets choix	a	b	c	d	e
init	0	$+\infty$	$+\infty$	$+\infty$	$+\infty$
a	/	5	3	$+\infty$	10
c	/	5	/	5	9
b	/	/	/	5	7
d	/	/	/	/	7
e	/	/	/	/	/

Arborescence des chemins ai-minimaux :



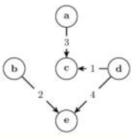
2. L'arborescence obtenue est-elle un arbre couvrant de poids minimal ? (donnez une justification détaillée).

Corrigé

(1.5 pts) L'arborescence est bien connexe et sans cycle et contient tous les sommets donc c'est un arbre couvrant. Il est de poids total 12.

Calculons un ACPM, on utilise l'algo de Kruskal : on sélectionne successivement (dc) poids 1, puis (be) poids 2, on ne peut pas prendre (cd) de poids 2 car ça créerait un cycle, (ac) poids 3, on ne peut pas prendre (ad) de poids 4 car ça créerait un cycle, on peut prendre (bc) ou (de) de poids 4.

On obtient un ACPM de poids 10 Donc l'arborescence de la question 1 n'est pas un ACPM puisque son poids est $12 > 10$.



3. Dessinez le graphe par niveaux si c'est possible sinon expliquez pourquoi ?

Corrigé

(0.5 pt) Il existe des circuits (par ex : cdc) donc on ne peut pas dessiner le graphe en niveaux.

4. Quel algorithme peut-on utiliser sur ce graphe pour calculer les chemins a-i-**Maximaux** Quel est le poids du chemin ae-maximal ?

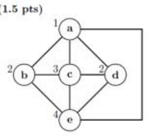
Corrigé

(1 pt) On peut utiliser Bellman-Kalaba ou Ford ou Floyd-Roy-Warshal, mais ni Dijkstra (pas valable pour maximaux) ni Bellman sans circuit (car circuits). Le chemin ae-maximal n'existe pas, car on peut passer une infinité de fois par le circuit (cdc) de poids 3.

5. Sans tenir compte du poids et des orientations des arcs, proposez une coloration du graphe. Montrez qu'on ne peut pas trouver moins que 3 couleurs.

Corrigé

(1.5 pts)



C'est une coloration à 4 couleurs. On ne peut pas trouver moins que 3 car il existe des cliques à 3 sommets (par ex : abc).

II. Routage (3 points)

Rappel : la **matrice des poids minimaux** associée à un graphe orienté [1] valué d'ordre n est une matrice de dimension $n \times n$ ayant pour termes λ_{ij} = longueur du chemin ij-minimal.
Rappel : la **matrice de routage** associée à un graphe orienté [1] valué d'ordre n est une matrice de dimension $n \times n$ ayant pour termes m_{ij} = sommet adjacent à i sur le chemin ij-minimal.

Après avoir fait tourner un algorithme de plus court chemin à partir d'un sommet, on réussit à compléter les matrices de routage et des poids minimaux de la façon suivante (les $\times$ signifient que la valeur n'est pas significative, les ? signifient qu'on ne peut pas déduire la valeur grâce aux résultats obtenus) :

$$\underbrace{\begin{pmatrix} \times & 3 & 3 & 3 & 3 & 3 \\ ? & \times & ? & ? & ? & ? \\ ? & 2 & \times & 4 & 5 & 4 \\ ? & ? & ? & \times & ? & 6 \\ ? & ? & ? & ? & \times & ? \\ ? & ? & ? & ? & ? & \times \end{pmatrix}}_{\text{Matrice de routage}} \quad \underbrace{\begin{pmatrix} \times & 4 & 2 & 0 & 7 & 4 \\ ? & \times & ? & ? & ? & ? \\ ? & 2 & \times & -2 & 5 & 2 \\ ? & ? & ? & \times & ? & 4 \\ ? & ? & ? & ? & \times & ? \\ ? & ? & ? & ? & ? & \times \end{pmatrix}}_{\text{Matrice des poids minimaux}}$$

1. À partir de ces matrices, donnez le nombre de sommets du graphe et dites si le graphe est orienté ou non (justifiez).

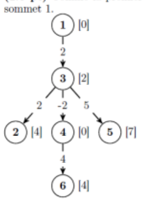
Corrigé

(0.5 pt) Il y a 6 sommets puisque c'est une matrice 6×6. Le graphe est orienté car la matrice des poids n'est pas symétrique (donc le plus court chemin dans un sens n'est pas le même que dans l'autre, ce n'est donc pas une chaîne mais un chemin différent dans un sens et dans l'autre).

2. Donnez le numéro du sommet à partir duquel on a lancé l'algorithme, et l'arborescence des chemins/chaînes minimaux/ales à partir de ce sommet.

Corrigé

(1.5 pt) Comme la première ligne est toute remplie on a calculé le chemin depuis le sommet 1.



3. On sait que l'arc ou l'arête (1,2) appartient au graphe, que peut-on dire sur son poids ?

Corrigé

(0.5 pt) Son poids est forcément supérieur ou égal à 4.

4. Listez tous les algorithmes de plus court chemin du cours qu'on a pu utiliser ?

Corrigé

(0.5 pt) pas Dijkstra (car poids négatif), on a pu utiliser Bellman sans circuit (si y avait pas de circuit), Bellman-Kalaba et Ford et Floyd-Roy-Warshall.

III. Ordonnancement MPM (5 points)

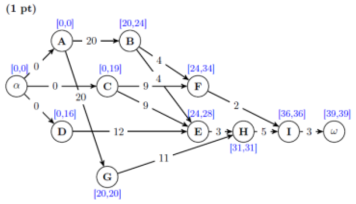
On considère le projet suivant décrit par les tâches et les contraintes entre tâches ci-dessous :

tâche	A	B	C	D	E	F	G	H	I
tâches requises	-	A	-	-	B, C, D	B, C	A	E, G	F, H
durée	20	4	9	12	3	2	11	5	3

1. Dessinez le graphe potentiels-tâches (MPM) correspondant à ce projet.

Corrigé

(1 pt)



2. Calculez la durée minimale du projet, les dates de début au plus tôt, les dates de début au plus tard, les marges totales, les marges libres [2] et les tâches critiques.

Corrigé

(2.5 pts) durée minimale 39, début au + tôt et au + tard : voir dessin

tâches	α	A	B	C	D	E	F	G	H	I	ω
marge totale	0	0	10	19	16	4	10	0	0	0	0
marge libre	0	0	0	15	12	4	10	0	0	0	0

tâches critiques (celles qui ont une marge totale nulle) : A, G, H, I.

3. Certaines tâches ont-elles des marges libres différentes de leur marge totale, pourquoi ?

Corrigé

(0.5 pt) B, C et D car suivies par des tâches qui ont de la marge totale, si elle prennent trop de temps sur leur marge elles peuvent empiéter sur la marge des tâches suivantes.

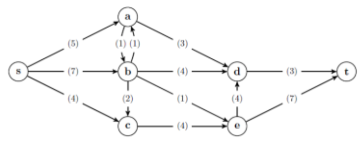
4. Sans tenir compte des durées des tâches, on désire trouver le nombre maximal de chemins *disjoints au sens des arcs* jusqu'à I. "Disjoints au sens des arcs" signifie que chaque arc peut être utilisé dans au plus un seul chemin entre le début du projet et I. Pour répondre à cette question, proposez un réseau de transport dans lequel le flot maximal donnera la solution. (**on ne demande pas de le résoudre**)

Corrigé

(1 pt) Il suffit de faire le même graphe en mettant des capacités de (1) sur tous les arcs sauf en sortie de (I) où on peut mettre $(+\infty)$, ainsi chaque arc ne sera utilisé qu'une seule fois dans une chaîne augmentante de α vers ω, sinon on excéderait les capacités.

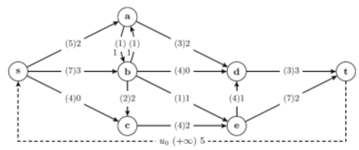
IV. Calcul de flot maximum (5 points)

On considère le réseau suivant où les arcs ont des capacités signalées entre parenthèses :



1. Montrez que le vecteur φ suivant est un flot compatible et donnez sa valeur.

Corrigé



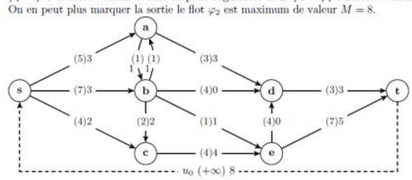
(1 pt) toutes les valeurs de φ sont compatibles : $\forall u, 0 \leq \varphi(u) \leq capa(u)$, d'autre part φ est un flot à condition de mettre 5 sur l'arc de retour, à ce moment là tout ce qui entre en chaque sommet = ce qui sort (loi de Kirchhoff). La valeur du flot est 5.

arcs	(s,a)	(s,b)	(s,c)	(a,b)	(a,d)	(b,a)	(b,c)	(b,d)	(b,e)	(c,e)	(e,d)	(d,t)	(e,t)
$\varphi(u)$	2	3	0	1	2	1	2	0	1	2	1	3	2

2. Donnez un flot maximum à l'aide de l'algorithme de Ford Fulkerson. Vous décrirez les différentes chaînes augmentantes utilisées et de combien vous avez augmenté à chaque étape. Vous décrirez ensuite ce flot maximum en donnant le flux sur chaque arc. Donnez la valeur du flot maximum (soit M cette valeur).

Corrigé

(2.5 pts) On part de φ, première chaîne augmentante : *sadet* on peut augmenter de 1, $\varphi_1 = \varphi + 1 sadets$: valeur 6. *scet* on peut augmenter de 2. $\varphi 2 = \varphi_1 + 2scets$: valeur 8. On en peut plus marquer la sortie le flot φ_2 est maximum de valeur $M = 8$.



Serait-il possible d'obtenir un flot de même valeur M sans utiliser l'arc (ad) ?

Corrigé

(0.5 pt) oui en faisant passer 3 de + sur (ab) et (bd) et 3 de moins sur (ad) et (sa).

Existe-t'il un arc dont l'augmentation de capacité de 1 permettrait de trouver un flot de valeur M+1 ? si oui : lequel ?, sinon : justifiez.

Corrigé

(1 pt) Les arcs dont les capacités empêchent d'augmenter le flot sont ceux qui sortent des derniers sommets marqués (sabcd) c'est à dire (dt) (be) et (ce). Il suffit donc d'augmenter de 1 la capa d'un de ces arcs.

Partiel 2020

1 Chemins optimums (6 points)

Soit G un graphe à 7 sommets dont les arcs (x, y) sont pondérés par p_{xy} décrit ci-dessous :

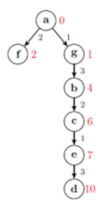
sommet x	a	b	c	d	e	f	g
$y \in \Gamma^+(x)$	b,f,g	c	d,e		d	e	b,c,e,f
p_{xy}	5,2,1	2	5,1		3	7	3,7,9,3

1. (**3 pts**) Calculez les chemins ai-minimaux dans le graphe G ci-dessous. Vous décrirez le déroulement de l'algorithme dans un tableau puis dessinerez l'arborescence des chemins ai-minimaux sur laquelle vous mentionnerez les distances de a à chaque sommet.

Corrigé

sommet choisi	a	b	c	d	e	f	g
init	0	∞	∞	∞	∞	∞	∞
a	/	5	∞	∞	∞	2	1
g	/	4	8	∞	10	2	/
f	/	4	8	∞	9	/	/
b	/	/	6	∞	9	/	/
c	/	/	/	11	7	/	/
e	/	/	/	10	/	/	/
d	/	/	/	/	/	/	/

On a entouré les poids des chemins ai-minimaux quand ils ont été obtenu pour la première fois. ça permet de savoir quel est le sommet qui a permis cette mise à jour. On obtient donc l'arborescence ci-contre.



2. (**1 pt**) Donnez le plus long chemin en termes de nombre d'arcs qui a une origine différente de a et qui est garanti minimal par le résultat de Dijkstra et grâce au principe de Bellman (tout chemin extrait d'un chemin minimal est minimal), donnez son poids.

Corrigé

Le plus long chemin ai-minimal en nombre d'arcs est $agbced$, tout chemin extrait de ce chemin est minimal, comme il ne doit pas avoir pour origine a c'est $gbced$ qui est le plus long chemin en nombre d'arcs garanti minimal, il est de poids 9.

3. (**2 pts**) L'arborescence obtenue est-elle un arbre couvrant de poids minimum ?

Corrigé

Pour obtenir un acpm on utilise l'algo de Kruskal : on sélectionne successivement les arcs (par ordre croissant des poids à condition qu'ils ne créent pas de cycle) :

arcs	(a,g)	(c,e)	(a,f)	(b,c)	(e,d)	(g,b)	STOP $n-1$ acs
poids	1	1	2	2	3	3	

L'acpm a pour poids 12, c'est le même poids que l'arborescence précédente qui est connexe et sans cycle, donc l'arborescence précédente est bien un acpm. (Notons que cette arborescence est même identique à l'acpm, notons qu'elle est unique car l'ajout de gf de poids 3 est impossible puisqu'il créérait un cycle avec deux autres arcs de poids inférieurs)).

2 Dijkstra et poids quelconques (3 pts)

Nous savons que l'algorithme de Dijkstra ne peut fonctionner que sur un graphe valué positivement alors que Bellman-Kalaba peut lui, être utilisé avec des arcs valués de façon quelconque.

Soit $G = (X, U)$ un graphe valué par une pondération p quelconque sur chaque arc, on note p_{xy} la pondération associée à l'arc (x, y). On décide de faire la procédure suivante :

- On pose $\quad m = \min_{(x,y)\in U} p_{xy}$
- On construit $G' = (X, U)$ identique au graphe G mais pondéré par $p'_{xy} = p_{xy} - m$

Pour les 3 questions suivantes, si la réponse est oui faites la démonstration sinon donnez un contre-exemple.

1. (**1 pt**) Peut-on montrer que G' est pondéré avec une pondération positive ou nulle sur chaque arc ?

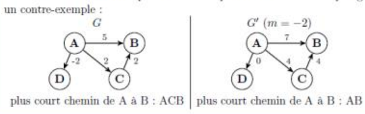
Corrigé

on note que $\forall (x, y) \in U, p_{xy} \geq m$ donc $\forall (x, y) \in U, p'_{xy} = p_{xy} - m \geq 0$. Donc G' est pondéré avec une pondération positive ou nulle sur chaque arc.

2. (**1 pt**) Peut-on montrer que tout chemin minimal de G' est un chemin minimal de G ?

Corrigé

Non car un chemin de n arcs aura un poids augmenté (quand m est négatif) de $-n \times m$. Donc si le chemin minimal possède beaucoup d'arcs il sera beaucoup augmenté. Voici un contre-exemple :



plus court chemin de A à B : ACB | plus court chemin de A à B : AB

3. (**1 pt**) Peut-on montrer que Dijkstra appliqué à G' donnera une arborescence de chemins qui sont minimaux aussi dans G ?

Corrigé

Non, Dijkstra calcule les plus court chemins dans un graphe pondéré positivement, donc il trouvera bien les plus court chemins de G', mais d'après la question précédente un plus court chemin de G' n'est pas forcément un plus court chemin de G.

3 Problème d'ordonnancement (5 points)

Lors de la construction d'une maison, on distingue 12 travaux distincts :

Tâche	Libellé de la tâche	Durée	Tâches à terminer avant
A	gros oeuvre	8	/
B	charpentes	2	A
C	toiture	1	B, A
D	plomberie	4	A
E	électricité	2	A
F	ravalement	3	A,B, C,D
G	fenêtre	1	A,B
H	aménagements extérieurs	1	C,D,E
I	plâtres	2	A, C,D,E,G
J	sols	2	D,E,G,I
K	peintures	3	I
L	emménagement	1	toutes les tâches

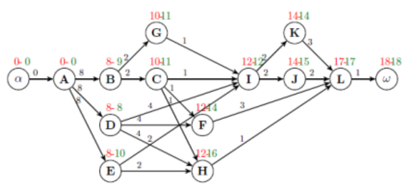
1. (**1 pt**) Y a-t-il des contraintes de précédence (données dans la dernière colonne du tableau) qui sont redondantes ? si oui donnez un exemple sinon prouvez-le.

Corrigé

oui par exemple C est après B et A or B doit être après A, il suffit donc de dire C après B.

2. (**2 pts**) Modélisez la situation à l'aide d'un graphe et déterminer la durée minimale du projet.

Corrigé



Durée minimale du projet : 18.

3. (**2 pts**) Indiquez sur votre graphe les dates de début au plus tôt et au plus tard de chaque tâche permettant de garantir cette durée optimale.

Corrigé

Voir dessin.

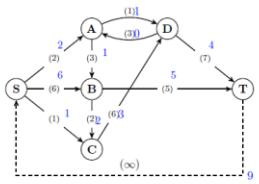
4 Problème de flots (6 points)

Le serveur S est connecté à la machine T par un réseau avec les noeuds A, B, C et D. Les capacités des débits maximums des connexions entre les noeuds sont données dans le tableau suivant (en Mbit/s). Les connexions sont orientées des sommets en ligne vers les sommets en colonne. Les cases vides signifient qu'il n'y a pas de connexion directe entre les noeuds. Les temps de passage dans les noeuds sont négligeables.

	A	B	C	D	T
S	2	6	1		
A		3		1	
B			2		5
C				6	
D	3				7

1. (**1 pt**) Dessinez le réseau. Est-ce un réseau de transport ?

Corrigé



oui en rajoutant l'arc de retour de (T) vers (S).

2. (**1 pt**) L'utilisateur de la machine T télécharge un très grand fichier du serveur S. On veut trouver quelle répartition des paquets sur les connexions permettra de maximiser le débit. Expliquer le lien avec la recherche d'un flot dans ce réseau. Quel type de flot recherche-t-on ?

Corrigé

On cherche à faire passer des informations de S à T de façon à ce qui entre en un noeud en sorte (pas de perte de paquet), on cherche donc un flot (vecteur qui vérifie la loi de Kirchhoff). D'autre part on ne peut pas avoir des paquets qui circulent à l'envers d'une liaison, ni un débit qui excède les capacités : on veut donc un flot compatible. Finalement on veut maximiser le débit c'est-à-dire qu'on veut qu'il arrive un maximum de paquets par seconde, on veut donc un vecteur qui maximise le nombre de données arrivant en T, donc un flot compatible de valeur maximale.

3. (**3 pts**) Trouvez un flot maximum en décrivant les chaînes augmentantes utilisées ainsi que les augmentations réalisées à chaque étape. Vous préciserez le nom de l'algorithme utilisé, vous prouverez que le flot est maximal.

Corrigé

On utilise l'algorithme de Ford-Fulkerson, on part du flot nul,
- on trouve une première chaîne augmentante SBT on peut augmenter de 5 sur cette chaine : $\varphi_0 = 5.[SBTS]$, flot de valeur 5.
- $SABCDT$ on peut augmenter de 2 : $\varphi_1 = \varphi_0 + 2.[SADTS]$ flot de valeur 7
- $SCDT$ on peut augmenter de 1 : $\varphi_2 = \varphi_1 + [SCDTS]$ flot de valeur 8.
- $SBADT$ on peut augmenter de 1 : $\varphi_3 = \varphi_2 + [SBADTS]$ flot de valeur 9
- on ne peut plus marquer que s le flot est donc max car on ne peut pas marquer t

4. (**1 pt**) Proposez une connexion dont l'augmentation de capacité permettrait d'augmenter le débit.

Corrigé

la connexion (sC)

Partiel 2021

I Ordonnancement (4 points)

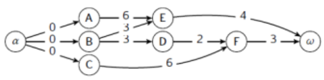
On considère un projet dont les tâches ont les durées et contraintes suivantes :

Taches	A	B	C	D	E	F
durées (en jours)	6	3	6	2	4	3
tâches prérequises				B	A,B	C,D

1. Dessinez le graphe potentiel-tâches.

Corrigé

1 pt



2. Faites un tableau récapitulant les dates de début au plus tôt, début au plus tard, marges totales, marges libres [1].

Corrigé

2 pts

	α	A	B	C	D	E	F	ω
t	0	0	0	0	3	6	6	10
t'	0	0	2	1	5	6	7	10
M	0	0	2	1	2	0	1	0
ml	0	0	0	0	1	0	1	0

3. Quelle est la durée du projet ? Quelles sont les tâches critiques ?

Corrigé

1 pt durée totale : 10 jours, taches critiques A et E.

II Chemins (7 points)

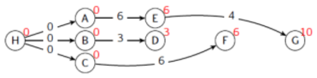
On considère le graphe G_1 suivant :

Sommets	H	A	B	C	D	E	F	G
$y \in \Gamma^+(x)$	A,B,C	E	D,E	F	F	G	G	
$poids(x,y)$	0,0,0	6	3,3	6	2	4	3	/

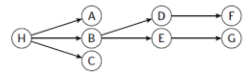
1. Calculez tous les chemins de poids MAXIMUM depuis le sommet H dans le graphe G_1. Vous donnerez le nom de l'algorithme utilisé et donnerez l'arborescence des chemins MAXIMAUX de racine H en écrivant à côté de chaque sommet X le poids du chemin HX-MAXIMAL y arrivant.

Corrigé

2 pts C'est le même graphe qu'à l'exercice précédent, et l'ordonnancement nous a déjà fait faire un bellman sans-circuit max :



2. L'algorithme de Dijkstra appliqué au sommet H donne l'arborescence suivante :



(a) Peut-on calculer le terme $\Lambda(B,F)$ de la matrice des longueurs [2] et le terme $M(B,F)$ de la matrice de routage[2], à partir de cette arborescence sans relancer d'algorithme ? Si oui donnez leurs valeurs et justifiez, sinon expliquez.

Corrigé

1 pt Le chemin de B à F est issu d'un chemin minimal (H,B,D,F) donc il est minimal.
Donc $\Lambda(B,F) = 3+2 = 5$ et M(B,F)=D.

(b) Même question pour les termes $\Lambda(E,F)$ et $M(E,F)$.

Corrigé

1 pt Le chemin de E à F n'est pas issu d'un chemin minimal dans l'arborescence des chemins minimaux par Dijkstra, donc on ne peut pas donner le chemin minimal de E à F sans relancer l'algo (notons qu'il n'existe pas).

3. Soit G_2 le graphe G_1 auquel on ajoute les arcs (E,H) et (G,C) :

Sommets	H	A	B	C	D	E	F	G
$y \in \Gamma^+(x)$	A,B,C	E	D,E	F	F	G, H	G	C

Calculez alors les composantes f-connexes de ce nouveau graphe G_2.

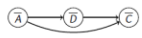
Corrigé

2 pts 3 comp. f-connexes $\overline{A} = \{A,B,E,H\}$, $\overline{D} = \{D\}$ et $\overline{C} = \{C,F,G\}$,

4. Donnez le graphe réduit de G_2 mis en niveau.

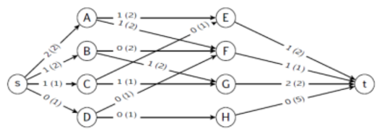
Corrigé

1 pt



III Calcul d'un flot Maximum (5 points)

On considère le graphe suivant où pour chaque arc on a donné la valeur d'un flot qui circule et la capacité de l'arc entre parenthèses.



1. Vérifiez que les valeurs données sur les arcs correspondent bien à un flot compatible, nommé φ_0, sur le réseau de transport associé à ce problème. Quel est la valeur du flot φ_0 ?

IV Équipements sportifs (SRI : 4 pts, STRI et L3-IRT : 2 pts)

Les joueuses du Stade Toulousain n'habitent pas toutes dans le même quartier de Toulouse. D'autre part, l'aglomération toulousaine est très étendue et terriblement embouteillée. Il n'est donc pas possible pour chaque joueuse de venir s'entraîner au stade toulousain tous les soirs. L'entraîneuse du Stade Toulousain Féminin (STF) a donc passé un accord avec la mairie de Toulouse pour que chaque quartier bénéficie d'un équipement sportif utilisable par les joueuses du STF et dont le STF financera en partie la construction.
Par contre, la mairie a exigé que des quartiers voisins n'aient pas le même équipement afin d'avoir un choix le plus large possible à proposer à la communauté (les élections

Corrigé

1.5 pts
- Flot : oui car vérifie la Loi de Kirchhoff en tout sommet ce qui entre=ce qui sort.
- Compatible : oui car pour tout arc u $0 \leq \varphi(u) \leq capa(u)$
- Valeur du flot :4

2. Calculez un flot maximum à partir de ce flot vous préciserez les chaînes augmentantes utilisées et de combien le flot est augmenté à chaque étape. Vous donnerez la valeur du flot maximum.

Corrigé

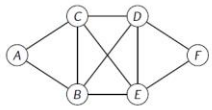
2.5 pts
- chaine augmentante sDHt, on augmente de 1 sur sDHts : $\varphi_1 = \varphi_0 + 1x(sDHts)$ valeur 5
- chaine augmentante sBFAEt, on augmente de 1 sur sBFAEts : $\varphi_2 = \varphi_1 + 1x(sBFAEts)$ valeur 6
- on ne peut plus marquer que s (donc on ne peut pas marquer pas la sortie) donc φ_2 est MAXIMUM.

3. (STRI et L3-IRT uniquement) Donnez une coupe de capacité minimum et donnez sa capacité.

Corrigé

1 pt coupe= $(s, ABCDEFGHt)$, sa capacité est $capa(c) = 2+2+1+1 = 6$ (confirme que φ_2 est maximum)

approchent, il faut que les gens soient satisfaits!). Le graphe suivant donne les proximités entre les quartiers A, B, C, D, E, F de la ville (une arête signifie que les quartiers sont voisins).



L'entraîneuse voudrait que ses joueuses de rugby s'entraînent soit sur un terrain de rugby, soit sur un stade d'athlétisme, soit dans une salle de boxe.
Est-ce qu'il est possible de construire seulement ces trois types d'équipement dans chacun des quartiers ? Si oui, montrer comment. Et si non, dîtes à l'entraîneuse combien de types d'équipements sportifs différents devront être utilisés au minimum par ses joueuses pour s'entraîner.
Dans tous les cas, il vous faudra traduire ce problème en termes de graphes puis le résoudre.
(SRI uniquement) Justifiez votre raisonnement et votre solution à l'aide des propriétés vues en TD.

Corrigé

C'est un problème de coloration :
- 1 sommet = 1 quartier,
- 1 arête= quartier voisin d'un autre quartier,
- 1 couleur = 1 type d'équipement sportif

En effet la contrainte : "deux quartiers voisins n'ont pas le même équipement" correspond à la contrainte d'une coloration dans un graphe "deux sommets adjacents n'ont pas la même couleur".
nb min de couleurs nécessaires : Il faut 4 couleurs au minimum car il existe une clique à 4 sommets B,C,D,E.
nb max de couleurs suffisantes : Le graphe est planaire (en traçant C vers E par l'extérieur), donc il est coloriable en moins de 4 couleurs. Autre démonstration, on trouve la coloration suivante avec 4 couleurs.
Il faut donc exactement 4 couleurs.
B=1, C=2, D=3, E=4, A=3 ou 4, F= 1 ou 2.