Problème 1 : Problème d'ordonnancement : La construction d'un stade.

Il s'agit d'un problème d'ordonnancement. Nous allons ajouter une tache fictive de durée nulle qui va représenter la date de fin du projet.

Les taches auront un indice de 1 à n, avec n désignant la tache fictive.

$$p_i \text{ dénote la durée de la tache } i$$

Afin de gérer la précédence entre les taches, nous allons nous appuyer sur un graphe de précédences G=(X,U) défini par un ensemble de taches X et un ensemble d'arcs U :

Un arc (i,j) signifie que la tache i doit précéder la tache j. (on peut facilement construire ce graphe à partir du tableau fourni).

Nous avons également besoin de variables de décision t_i pour les dates de début des taches, comptées à partir du temps zéro.

Les seules contraintes à respecter sont les contraintes de précédence : s'il existe un arc de i vers j, alors la date de fin de $i\ (t_i + p_i)$ ne doit pas dépasser la date de début de j.

La formulation de la contrainte :

$$\forall (i,j) \in U : t_i + p_i \ \leq \ t_j$$

La fonction objectif :

$Min\ t_n$ (minimiser la date de début de la tache fictive veut dire minimiser la date de fin du projet, puisque toutes les taches devront être finies avant le début de la tache fictive).

Enfin bien sûr : $\forall\ i = 1, \dots, n :\ t_i\ \geq 0$

En résumé vous devez implémenter la fonction objectif ainsi que les contraintes suivantes :

$$\boldsymbol{Min\ t_n}$$

$$\forall (\boldsymbol{i}, \boldsymbol{j}) \in \boldsymbol{U} : \boldsymbol{t_i} + \boldsymbol{p_i} \ \leq \ \boldsymbol{t}_{\mathrm{j}}$$

$$\forall\ \boldsymbol{i} = \boldsymbol{1}, \dots, \boldsymbol{n} :\ \boldsymbol{t_i}\ \geq \boldsymbol{0}$$

Problème 2 affectation du personnel à des postes :

Ceci est un problème d'affectation simple :

Pour modéliser ce problème facilement, il suffit de défini des variables de décision de type binaire. Nous allons avoir n^2 $variables$ binaires, notées $\boldsymbol{x_{ij}}$.

Nous allons également noter $\boldsymbol{p_{ij}}$ **la productivité de la personne i au poste j**

$x_{ij} = 1$ Veut dire que la personne i est affectée à la machine j.

x_{ij} = 0 sinon.

Nous allons également noter « n » comme le nombre de machines et de personne dans l'usine.

La modélisation du problème se fera comme ceci :

$$\forall\ i = 1, \dots, n, \forall\ j = 1, \dots, n :\ x_{ij} \in \{0,1\}$$

$$\forall\ i = 1, \dots, n :\ \sum_{j=1}^{n} x_{ij} = 1\ (une\ personne\ est\ affect\acute{e}e\ \text{à}\ un\ seul\ poste$$

$$\forall\ j = 1, \dots, n :\ \sum_{i=1}^{n} x_{ij} = 1\ (un\ poste\ re\c{c}oit\ une\ seule\ personne)$$

La fonction objectif :

$$Max\ \sum_{i=1}^{n}\sum_{j=1}^{n} p_{ij}x_{ij}$$

Problème 3 :

Problème d'adduction d'eau :

Le problème décrit dans cet exercice est un problème de flot maximum dans un réseau de transport (vous avez vu ça en théorie des graphes).

Nous allons d'abord créer une source et un puit dans le graphe (voir votre cours de TG de l'année passée).

Le nœud source sera le nœud 11, relié au réservoirs par deux arcs dont les capacités sont égales aux disponibilité des deux réservoirs (voir le graphe ci-dessou).

Le nœud puit sera le nœud 12 (voir le graphe ci-dessous) auquel les villes sont reliées par trois arcs, dont les capacités sont égales aux besoins des villes.

Le graphe résultant est le graphe G=(X,A,C,s,t) avec :

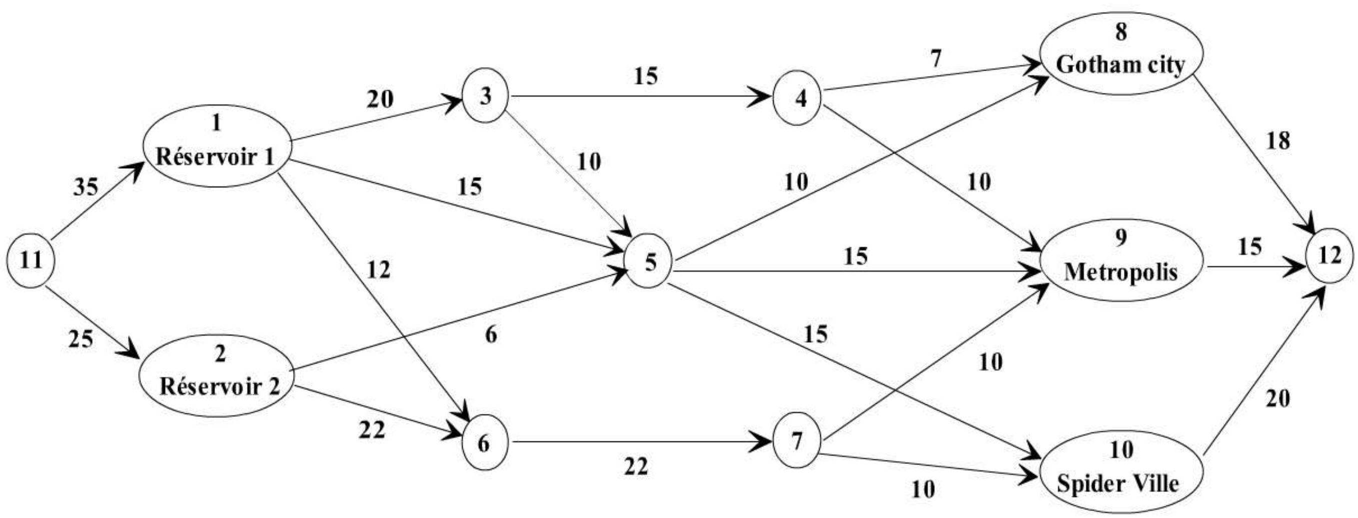
X représentant l'ensemble des sommets.

A représentant l'ensemble des arcs

C désignant la capacité de l'arc (i,j).

S est la source du réseau (le nœud 11 dans le graphe).

T est le puit du réseau (nœud 12).



Un flot de débit F dans ce réseau est une application Φ de U vérifiant les contraintes suivantes :

$$(1) \quad \forall (i,j) \in A : \Phi_{ij} \leq C_{ij}$$

$$(2) \quad \forall i \neq s,t : \sum_{j \in \Gamma(i)} \Phi_{ij} = \sum_{j \in \Gamma^{-1}(i)} \Phi_{ji}$$

$$(3) \quad F = \sum_{j \in \Gamma^{-1}(t)} \Phi_{jt}$$

$$(4) \quad \forall (i,j) \in A : \Phi_{ij} \geq 0$$

La contrainte (1) indique que le flot ne doit pas excéder la capacité de l’arc

La contrainte (2) indique que le flux entrant dans chaque arc, doit absolument en sortir.

La contrainte (3) indique que le débit total F traversant ce réseau, est égale au flot sortant de s.

La contrainte (4) indique qu’un flot ne peut jamais être négatif.

La fonction objectif est bien sûr : Max F, pour connaitre le débit total permis dans ce réseau.

Problème 4: Surveillance des rues par des caméras :

Pour résoudre cet exercice, il va d’abord falloir convertir la carte en un graphe non orienté G=(X,R).

L’ensemble X représente l’ensemble des emplacement des caméras.

L’ensemble R représente les rues entre les emplacements.

Nous allons définir la variable de décision x_i qui vaut 1 si une caméra est positionnée à l’emplacement i et 0 sinon.

Après avoir modélisé le graphe, la modélisation du problème est très simple :

$$(1) \qquad \text{Min} \sum_{i=1}^{n} x_i$$

$$(2) \quad \forall\,(i,j) \in R : x_i + x_j \geq 1$$

$$(3) \quad \forall\, i = 1 \ldots n : x_i \in \{0,1\}$$

La contrainte (2) signifie que chaque rue doit être surveillée.

Problème 5 : placement des perceptions :

Pour résoudre ce problème, nous aurons besoin d'une matrice D ci-dessous, qui va nous fournir la distance entre chaque ville

	1	2	3	4	5	6	7	8	9	10	11	12
1	0	15	37	55	24	60	18	33	48	40	58	67
2	15	0	22	40	38	52	33	48	42	55	61	61
3	37	22	0	18	16	30	41	28	20	58	39	39
4	55	40	18	0	34	12	59	46	24	62	43	34
5	24	38	16	34	0	36	25	12	24	47	37	43
6	60	52	30	12	36	0	57	42	12	50	31	22
7	18	33	41	59	25	57	0	15	45	22	40	61
8	33	48	28	46	12	42	15	0	30	37	25	46
9	48	42	20	24	24	12	45	30	0	38	19	19
10	40	55	58	62	47	50	22	37	38	0	19	40
11	58	61	39	43	37	31	40	25	19	19	0	21
12	67	61	39	34	43	22	61	46	19	40	21	0

Matrice D

Nous allons noter H_i le nombre d'habitants de la ville i.

Nous allons avoir également besoin de deux groupes de variables binaires :

y_j qui vaut 1 si et seulement si une perception est installée dans la ville j .

x_{ij} qui vaut 1 si la ville i dépend de la perception j. Ces variables x_{ij} vont permettre de calculer la distance moyenne à parcourir par habitant et de préciser la perception dont chaque ville va dépendre.

(1) $\forall\, j = 1, \dots, 12 \quad y_j \in \{0,1\}$
(2) $\forall\, j = 1, \dots, 12\, , \forall\, i = 1, \dots, 12 :\ x_{ij} \in \{0,1\}$

(3) $\sum_{j=1}^{12} y_j = 3$ (Puisque nous devons avoir 3 perceptions)

(4) $\forall\, i = 1, \dots, 12 :\ \sum_{j=1}^{12} x_{ij} = 1$ (Cette contrainte signifie que chaque ville ne devra dépendre que d'une seule perception).

La fonction objectif : $Min\ \sum_{i=1}^{12} \sum_{j=1}^{12} H_i D_{ij} x_{ij}$

Les étudiants devrons fournir la valeur de la fonction objectif lors de la soumission.

Problème 6 : Correspondance d'avions :

Soit n, le nombre d'avions et P_{ij} le nombre de passagers en transfert entre i et j .

Variables de décision : x_{ij} = 1 si est seulement si l'avion en provenance de i assure le vol à destination de j.

On veut minimiser le nombre de passagers changeant d'avion, mais ceci est équivalent à l'objectif consistant à maximiser le nombre de passagers restant dans l'avion.

La fonction objectif : $Max\ \sum_{i=1}^{n} \sum_{j=1}^{n} P_{ij} x_{ij}$

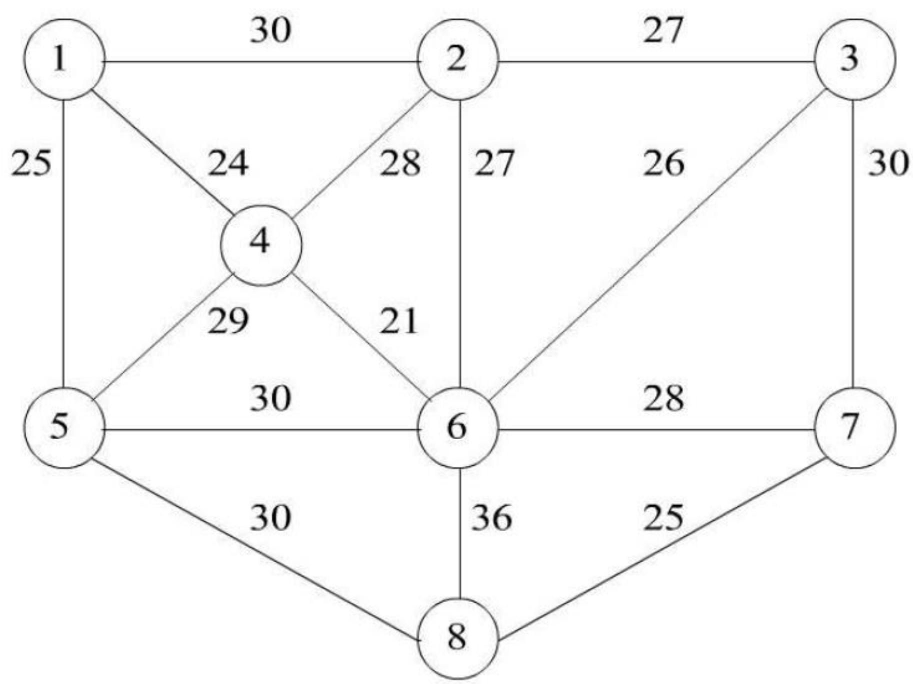
Avec les contraintes :

(1) $\forall\, j = 1, \dots, n :\ \sum_{i=1}^{n} \boldsymbol{x_{ij}} = 1$ une ville de destination ne peut être desservie que par un seul avion.
(2) $\forall\, i = 1, \dots, n :\ \sum_{j=1}^{n} x_{ij} = 1$ Une ville d'origine ne peut desservir qu'une seule ville de destination.
(3) $\forall\, i = 1, \dots, n, \forall\, j = 1, \dots, n :\ x_{ij} \in \{0,1\}$.

Problème 7 constitution d'un équipage :

Nous allons noter np le nombre de pilote.

Nous allons modéliser ce problème avec un graphe de compatibilité G(X,E), non orienté. X est un ensemble de np nœuds correspondant aux pilotes. Une arrête (i,j) existe entre deux nœuds i et j si et seulement si les deux pilotes sont compatibles. C'est-à-dire qu'ils ont une langue et un avion commun pour lesquels ils ont au moins 10/20. L'arrête est valuée pas un poids égale au score c_{ij} atteint par l'équipage. Voici ci-dessous le graphe résultant :



Graphe de compatibilité entre les pilotes

Modélisation du problème :

Nous allons utiliser une variable binaire $\boldsymbol{x_{ij}}$ pour indiquer qu'une arrête du graphe est prise ou pas. (Une arrête prise dans le graphe veut dire que les deux pilotes adjacents à cette arrête vont constituer un même équipage).

$$(1) \quad \text{Max} \sum_{[i,j]\in E} c_{ij} x_{ij}$$

$$(2) \quad \forall k \in X : \sum_{[i,j]\in E,\, i=k \text{ ou } j=k} x_{ij} \leq 1$$

$$(3) \quad \forall [i,j] \in E : x_{ij} \in \{0,1\}$$

Nous allons maximiser avec la fonction objectif le cumul des poids choisis.

La contrainte (2) indique que tout nœud k a au plus une seule arrête incidente. C'est un dire qu'un pilote ne peut faire partie, que d'un seul équipage au maximum.

Problème 8 :

Notons *nville* le nombre de villes et *nhub* le nombre de hubs. Soit D_{ij} la distance entre deux villes *i* et *j* et Q_{ij} la quantité de fret à transporter de *i* vers *j*. Le coût de transport d'une tonne de fret de *i* vers *j* dépend des hubs sélectionnés. En effet, si *k* et *m* sont des hubs alors le fret à transporter de toute ville *i* vers toute ville *j* va transiter par *k* et *m*.

Notons C_{ijkm} le coût de transport de *i* à *j* si *k* et *m* sont des hubs. Ce coût est égal au coût de transport de *i* à *k*, plus le coût de transport de *k* à *m*, plus le coût de transport de *m* à *j*. Et le coût de *k* à *m* correspond à α =80% du coût normal de *k* à *m* renseigné au puisqu'il s'agit d'un coût de transport inter-hub.

Soit la variable $flow_{ijkm}$ égale à 1 si le fret transporté de *i* à *j* passe par les hubs *k* et *m* dans cet ordre, 0 sinon. Et soit la variable hub_i valant 1 si *i* est un hub, 0 sinon. Le programme linéaire en nombres entiers modélisant ce problème est donné ci-dessous.

La fonction objective (1) minimise le total des coûts de transport. La contrainte (2) indique que l'on veut exactement *nhub* =2 hubs. La contrainte (3) impose que chaque paire de villes (i, j) soit affectée à une et une seule paire de hubs. Les contraintes (4) et (5) impliquent que si la variable $flow_{ijkm}$ vaut 1, alors les variables hub_k et hub_m valent 1. En d'autres termes, du fret devant être transporté de *i* à *j* ne peut transiter par *k* et *m si k* et *m* sont des hubs. Pour terminer, les contraintes (6) et (7) indiquent que les variables sont binaires.

Remarque :

Dans la contrainte (3), *k* peut être égal à *m*. On autorise donc du fret à ne transiter que par un seul hub. C'est le cas lorsque les deux villes origine/destination sont affectées au même hub. Le cout de transport inter-hub $\alpha\, c_{kk} = \alpha\, c_{km}$ est alors nul.

$$(1)\quad Min \sum_{i=1}^{n}\sum_{j=1}^{n}\ \sum_{k=1}^{n}\sum_{m=1}^{n} Ci_{jkm} \cdot Q_{ij} \cdot flow_{ijkm}$$

$$(2) \sum_{k=1}^{n} hub_k = p$$

$$(3) \forall i = 1..n,\ j = 1..n : \sum_{k=1}^{n}\sum_{m=1}^{n} flow_{ijkm} = 1$$

$$(4)\quad \forall i = 1..n,\ j = 1..n, \forall k = 1..n, m = 1..n :\ flow_{ijkm} \leq hub_k$$

$$(5)\quad \forall i = 1..n,\ j = 1..n, \forall k = 1..n, m = 1..n :\ flow_{ijkm} \leq hub_m$$

$$(6)\quad \forall k = 1..n : hub_k \in \{0,1\}$$

$$(7)\quad \forall i = 1..n,\ j = 1..n, \forall k = 1..n, m = 1..n :\ flow_{ijkm} \in \{0,1\}$$