Problème d'affectation

+ n personnes numérotées de 1 à n
+ n tâches à réaliser numérotées de 1 à n
+ c_{ij} pour i allant de 1 à n et j allant de 1 à n : coût si la personne i fait la tâche j.

BUT : Donner à chaque personne une seule tâche à réaliser pour minimiser le coût total.

Représentation par matrice :

tâches

$$
M = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array}
\begin{array}{c}
\begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\
\begin{bmatrix} & & & \\ & & & \\ & c_{32} & & \\ & & & \end{bmatrix}
\end{array}
$$

personnes

c_{32} $\in \mathbb{N}$: coût si la personne 3 fait la tâche 2

Affectation

$$
\begin{array}{c|cccc}
 & 1 & 2 & 3 & 4 \\
\hline
1 & & & \square & \\
2 & \square & & & \\
3 & & \square & & \\
4 & & & & \square
\end{array}
$$

Une case par ligne et une case par colonne.

Propriétés :

Si l'on a deux matrices M et M' de même taille et que M' est obtenu à partir de M en enlevant (ou ajoutant) une même quantité c sur une ligne, la meilleure affectation pour M est la même que pour M'.
(mais le coût peut changer)

Cas simple

$$M : \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 1 & 0 & 1 & 3 \\ 2 & 2 & 4 & 0 & 1 \\ 3 & 1 & 6 & 3 & 0 \\ 4 & 0 & 0 & 4 & 5 \end{array}$$

→ Dans M, on peut trouver n zéros tels que aucun de ces zéros n'est sur la même ligne ou sur la même colonne qu'un autre 0.

→ Donne automatiquement une affectation de coût 0

$$M : \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 1 & \boxed{0} & 1 & 3 \\ 2 & 2 & 4 & \boxed{0} & 1 \\ 3 & 1 & 6 & 3 & \boxed{0} \\ 4 & \boxed{0} & 0 & 4 & 5 \end{array}$$

Algorithme hongrois : Se ramener à ce cas là

Algorithme hongrois (Algorithme de Kuhn-Munkres)

$$\Pi = \begin{bmatrix} 1500 & 4000 & 4500 \\ 2000 & 6000 & 3500 \\ 2000 & 4000 & 2500 \end{bmatrix}$$

Étape 1 : Dans chaque ligne, on enlève à chaque élément le minimum de la ligne.

$$\Pi = \begin{bmatrix} 0 & 2500 & 3000 \\ 0 & 4000 & 1500 \\ 0 & 2000 & 500 \end{bmatrix}.$$

$$M = \begin{bmatrix} 0 & 2500 & 3000 \\ 0 & 4000 & 1500 \\ 0 & 2000 & 500 \end{bmatrix}$$

<u>Étape 2</u> : Dans chaque colonne, on enlève à chaque élément le minimum de la colonne.

$$M = \begin{bmatrix} 0 & 500 & 2500 \\ 0 & 2000 & 1000 \\ 0 & 0 & 0 \end{bmatrix}$$

$$M = \begin{bmatrix} 0 & 500 & 2500 \\ 0 & 2000 & 1000 \\ 0 & 0 & 0 \end{bmatrix}$$

<u>Étape 3</u> : En traçant des lignes verticales ou horizontales, recouvrir tous les 0 avec un minimum de ligne (on verra comment faire cela plus tard !).

$$M = \begin{bmatrix} 0 & 500 & 2500 \\ 0 & 2000 & 1000 \\ 0 & 0 & 0 \end{bmatrix}$$

Si le nombre minimal est n (le nombre de lignes de la matrice → on a une affectation optimale → FIN.

Sinon ⇝ <u>Étape 4</u>

$$M = \begin{bmatrix} 0 & 500 & 2500 \\ 0 & 2000 & 1000 \\ 0 & 0 & 0 \end{bmatrix}$$

<u>Étape 4</u> : Retirer à toutes les lignes non barrées le plus petit entier non barré et l'ajouter aux colonnes barrées

$$M = \begin{bmatrix} -500 & 0 & 2000 \\ -500 & 1500 & 500 \\ 0 & 0 & 0 \end{bmatrix}$$

$$M = \begin{bmatrix} 0 & 0 & 2000 \\ 0 & 1500 & 500 \\ 500 & 0 & 0 \end{bmatrix}$$

$$M = \begin{bmatrix} 0 & 0 & 2000 \\ 0 & 1500 & 500 \\ 500 & 0 & 0 \end{bmatrix}$$

Retour à l'étape 3 :

$$M = \begin{bmatrix} 0 & \boxed{0} & 2000 \\ \boxed{0} & 1500 & 500 \\ 500 & 0 & \boxed{0} \end{bmatrix}$$

↳ il y a 3 lignes au minimal pour couvrir les 0

↳ Assignement minimal

$$\Pi = \begin{bmatrix} 0 & \boxed{0} & 2000 \\ \boxed{0} & 1500 & 500 \\ 500 & 0 & \boxed{0} \end{bmatrix}$$

↳ On revient à la matrice initiale.

$$\Pi = \begin{bmatrix} 1500 & \boxed{4000} & 4500 \\ \boxed{2000} & 6000 & 3500 \\ 2000 & 4000 & \boxed{2500} \end{bmatrix}$$

↳ coût minimal : 8500

Retour sur l'étape 3

Étape 3 : En traçant des lignes verticales ou horizontales, recouvrir tous les 0 avec un minimum de ligne

$$\begin{bmatrix} 12 & 9 & 4 & 0 & 7 \\ 11 & 10 & 5 & 0 & 5 \\ 7 & 9 & 9 & 6 & 0 \\ 0 & 3 & 10 & 13 & 9 \\ 5 & 0 & 0 & 4 & 9 \end{bmatrix}$$

On va barrer et encadrer des 0 dans cette matrice à 5 lignes et 5 colonnes

$\boxed{3.1}$. a) Chercher la ligne avec le moins de zéros non barrés (si il y en a plusieurs, on prend la plus haute)

b) Encadrer un zéro (non barré) de cette ligne

c) Barrer tous les zéros se trouvant sur la même ligne ou sur la même colonne que le zéro encadré

↳ On répète jusqu'à ce que tous les zéros soient barrés ou encadrés

$$\begin{bmatrix} 12 & 9 & 4 & \boxed{0} & 7 \\ 11 & 10 & 5 & \cancel{0} & 5 \\ 7 & 9 & 9 & 6 & \boxed{0} \\ \boxed{0} & 3 & 10 & 13 & 9 \\ 5 & \boxed{0} & \cancel{0} & 4 & 9 \end{bmatrix}$$

$$\begin{bmatrix} 12 & 9 & 4 & \boxed{0} & 7 \\ 11 & 10 & 5 & \cancel{0} & 5 \\ 7 & 9 & 9 & 6 & \boxed{0} \\ \boxed{0} & 3 & 10 & 13 & 9 \\ 5 & \boxed{0} & \cancel{0} & 4 & 9 \end{bmatrix}$$

$\boxed{3.2}$

a) Marquer d'une croix toutes les lignes avec aucun zéro encadré

b) Marquer d'une croix les colonnes ayant un zéro barré sur une ligne marquée

c) Marquer d'une croix les lignes ayant un zéro encadré dans une colonne marquée

Répéter b) et c) jusqu'à ne plus pouvoir mettre de croix.

$$
\begin{array}{c}
\begin{bmatrix}
12 & 9 & 4 & \boxed{0} & 7 \\
11 & 10 & 5 & \cancel{0} & 5 \\
7 & 9 & 9 & 6 & \boxed{0} \\
\boxed{0} & 3 & 10 & 13 & 9 \\
5 & \boxed{0} & \cancel{0} & 4 & 9
\end{bmatrix}
\begin{array}{c} \times \\ \times \\ \\ \\ \\ \end{array} \\
\begin{array}{ccccc} & & & \times & \end{array}
\end{array}
$$

$$
\begin{array}{cc}
\begin{bmatrix}
12 & 9 & 4 & \boxed{0} & 7 \\
11 & 10 & 5 & \cancel{0} & 5 \\
7 & 9 & 9 & 6 & \boxed{0} \\
\boxed{0} & 3 & 10 & 13 & 9 \\
5 & \boxed{0} & \cancel{0} & 4 & 9
\end{bmatrix}
&
\begin{array}{c} X \\ X \\ \\ \\ \end{array}
\\
\begin{array}{ccccc} & & & X & \end{array}
&
\end{array}
$$

$\boxed{3.3}$ On trace une ligne sur toute ligne non marquée et toute colonne marquée.

Lien avec le TPG :

- N ingrédients
- n livreurs
- 1 restaurant

Point clef : Chaque livreur peut prendre un ingrédient à la fois pour l'amener au restaurant et repartir ensuite.

↝ Équivalent à avoir

→ $M + (N-1)$ livreurs (les $(N-1)$ nouveaux livreurs sont au restaurant)

→ $N + (n-1)$ ingrédients (les $(n-1)$ nouveaux ingrédients ont un coût 0 de transport)

→ Problème d'affectation