

Problème type : on cherche à affecter n personnes dans n postes différents, chaque personne ayant classé par ordre de préférence chacun des postes, le but étant d'obtenir avec un algorithme rapide une solution maximisant la satisfaction globale des personnes, définie comme la somme des satisfactions individuelles.

I. Problème posé (R. Faure)

Une administration veut affecter $n = 5$ personnes salariées "A, B, C, D, E" aux postes "a, b, c, d, e" en maximisant la satisfaction générale. Les classements des postes sont donnés par la matrice c_{ij} de taille $n \times n$ (tableau I). c_{ij} (ligne i , colonne j) donne le classement du poste j par la personne i .

	a	b	c	d	e
A	1	2	3	4	5
B	1	4	2	5	3
C	3	2	1	5	4
D	1	2	3	5	4
E	2	1	4	3	5

tableau I : classement des postes

Exemple d'affectation : A-a, B-c, C-b, D-e, E-d pour un coût total de $1 + 2 + 2 + 4 + 3 = 12$.

- *But* : le problème consiste à choisir une case et une seule par ligne et par colonne de façon à minimiser la somme des chiffres des $n = 5$ cases choisies.
- *Formulation mathématique* : (on pose $x_{ij} = 1$ si la personne i obtient le poste j , 0 sinon)

On cherche à déterminer les x_{ij} pour minimiser le coût total $\sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij}$ sous les

contraintes $\sum_{i=1}^n x_{ij} = 1$ (le poste j est affecté à une seule personne) et $\sum_{j=1}^n x_{ij} = 1$ (la personne i est affectée à un seul poste).

- *Remarque* : si les personnes avaient noté les postes par une note n_{ij} de 0 (mauvais poste) à 20

(poste excellent), il faudrait maximiser $\sum_{i=1}^n \sum_{j=1}^n n_{ij} x_{ij}$ sous les mêmes contraintes. On revient

à un problème de minimisation (et on peut alors appliquer la méthode hongroise) en remplaçant chaque note n_{ij} par la note $n'_{ij} = 20 - n_{ij}$.

II. Résolution par la méthode hongroise (Kuhn, Egervary, König)

Le but du projet TAI est de résoudre ce type de problème par la méthode hongroise.

- 1) A l'aide de vos notes de cours, d'une recherche sur internet ou dans la littérature (l'accès à la Bibliothèque Mathématiques - Informatique Enseignement de Jussieu, Patio 46-55 niveau Jussieu, est libre, vous ne pouvez pas emprunter d'ouvrages mais des photocopieuses sont disponibles, renseignements sur le site <http://www.bius.jussieu.fr/> de la Bibliothèque Interuniversitaire Scientifique de Jussieu), vous devrez en particulier :

- distinguer 3 phases dans le déroulement de l'algorithme (obtention initiale de zéros, recherche d'une solution de coût nul, recherche d'une solution optimale),
- déterminer comment obtenir le nombre de solutions optimales de coût équivalent,
- préciser la vitesse de convergence de l'algorithme.

- 2) Résoudre le problème posé ci-dessus. Déterminer le nombre de solutions équivalentes.
- 3) Résoudre le double problème d'affectation suivant : le tableau I donne les préférences de poste des 5 personnes salariées. De leur côté les employeurs ont étudié le dossier de ces 5 personnes, ce qui leur a permis d'établir l'ordre de préférence de ces 5 personnes pour chacun des postes. Cet ordre de préférence est donné dans le tableau II. Déterminer quelle est la solution optimale qui maximise la satisfaction globale "employeurs + salariés" et quel est le nombre de solutions équivalentes.

	A	B	C	D	E
a	2	1	5	4	3
b	2	3	1	4	5
c	1	4	5	3	2
d	2	4	1	3	5
e	1	2	3	5	4

tableau II : classement des personnes

Travail à fournir :

- le travail sera effectué par groupe de 3 personnes (exceptionnellement 4) faisant **impérativement** partie du même groupe de TD : les délégués seront chargés de communiquer ces groupes aux enseignants ainsi qu'à l'administration de l'EFREI.
- un rapport écrit exposant clairement vos résultats (10 pages maximum) sera rendu par chaque groupe une semaine avant la date de soutenance.
- un exposé oral sera préparé. Sa durée, l'ordre de passage ainsi que la salle de soutenance vous seront communiqués ultérieurement. Moyens pour la présentation : tableau, rétroprojecteur ou vidéoprojecteur (moyen recommandé). Vous devrez apporter un ordinateur contenant la présentation.
- un logiciel (écrit en C ou C++) sera réalisé pour répondre aux questions 2 et 3 ; afin de tester ce logiciel pendant la présentation, le logiciel devra lire les paramètres du problème d'affectation dans un fichier de configuration ASCII (extension .txt) dont la structure est indiquée plus bas.
- afin de pouvoir noter les postes et les personnes au lieu simplement de les classer, on complètera le fichier de configuration par la donnée des notes de la façon indiquée ci-après (meilleure note 20, moins bonne note 0).
- *structure du fichier de configuration :*

ligne 1 : n (nombre de personnes)

ligne 2 : *type_pbm* (1 pour affectation simple sans notes, 2 pour affectation double sans notes, 3 pour affectation simple avec notes, 4 pour affectation double avec notes)

ligne 3 : $a_1 a_2 \dots a_n$ (numéros des postes de la personne 1 classés par ordre décroissant)

ligne 4 : $b_1 b_2 \dots b_n$ (numéros des postes de la personne 2 classés par ordre décroissant)

...

ligne $n+2$: $n_1 n_2$ n_n (numéros des postes de la personne n classés par ordre décroissant)
 ligne $n+3$: $A_1 A_2$ A_n (numéros des personnes classés par ordre décroissant pour le poste 1)
 ligne $n+4$: $B_1 B_2$ B_n (numéros des personnes classés par ordre décroissant pour le poste 2)
 ...
 ligne $2n+2$: $N_1 N_2$ N_n (numéros des personnes classés par ordre décroissant pour le poste n)
 n lignes suivantes (si nécessaire) : les notes des postes pour chaque personne
 n lignes suivantes (si nécessaire) : les notes des personnes pour chaque poste.

Le fichier de configuration sera donc :

- "config_Q2.txt" pour la question 2 (affectation simple sans notes) :

```

5
1      affectation simple sans notes
1 2 3 4 5  attention ces 5 lignes ressemblent au tableau I mais c'est un hasard !
1 3 5 2 4  La deuxième ligne est différente de celle du tableau I.
3 2 1 5 4  Il faut donc reconstruire ces 5 lignes à partir du tableau I et pas
1 2 3 5 4  simplement recopier le tableau I
2 1 4 3 5
  
```

- "config_Q3.txt" pour la question 3 (affectation double sans notes) :

```

5
2      affectation double sans notes
1 2 3 4 5  }
1 3 5 2 4  } les numéros des postes donnés par les 5 personnes
3 2 1 5 4  }
1 2 3 5 4  }
2 1 4 3 5  }
2 1 5 4 3  }
3 1 2 4 5  } les numéros des personnes donnés pour les 5 postes
1 5 4 2 3  }
3 1 4 2 5  }
1 2 3 5 4  }
  
```

- "config_Q2n.txt" pour la question 2 (affectation simple avec notes) :

```

5
3      affectation simple avec notes
1 2 3 4 5  }
1 3 5 2 4  } les numéros des postes donnés par les 5 personnes
3 2 1 5 4  }
1 2 3 5 4  }
2 1 4 3 5  }
20 18 10 5 0  } les notes attribuées
20 15 10 5 4  } aux postes par les
15 14 13 2 1  } 5 personnes
20 15 14 13 12  (tableau ci-contre)
20 3 2 1 0
  
```

	a	b	c	d	e
A	20	18	10	5	0
B	20	5	15	4	10
C	13	14	15	1	2
D	20	15	14	12	13
E	3	20	1	2	0

tableau I : notes des postes

Le logiciel devra permettre la résolution de problèmes de grande dimension, on se limitera néanmoins à la taille maximale $n = 300$. Pour éviter d'avoir à entrer tous les classements des postes et personnes dans ces problèmes à grande dimension, on utilisera le lettre "f" (pour "fin") pour dire que le classement est terminé. Ainsi si la personne A classe les postes dans l'ordre "a b c" et ne classe pas les postes "d" et "e", la troisième ligne du fichier "config_Q2" sera " 1 2 3 f". Bien entendu, on attribuera alors un très grand nombre M aux classements des postes "d" et "e" pour la personne A pour éviter que l'un de ces postes ne lui soit attribué.

Le code source ainsi que l'algorithme devront être présentés pendant la soutenance orale. Afin de pouvoir vérifier le résultat du code sur des problèmes de grande dimension, on stockera les résultats dans un fichier texte, de nom "resultats.txt", comprenant un nombre par ligne dans l'ordre suivant :

valeur minimale obtenue pour la satisfaction globale
nombre N de solutions équivalentes
numéro du poste attribué à la personne A dans la solution 1
numéro du poste attribué à la personne B dans la solution 1
...
numéro du poste attribué à la personne A dans la solution 2 (si elle existe)
numéro du poste attribué à la personne B dans la solution 2 (si elle existe)
...
numéro du poste attribué à la personne A dans la solution N (si elle existe)
numéro du poste attribué à la personne B dans la solution N (si elle existe)
...