

Affectation

Travaux Pratiques

Dans ce TP nous allons programmer en python l'algorithme hongrois comme nous l'avons vu en TD.

A cette fin, le travail préparatoire demandé, l'écriture de la bibliothèque **GrapheXy** est nécessaire.

Les variables globales de la programmation de l'algorithme hongrois sont :

G qui est une matrice carré représentant le graphe réduit augmenté par la valuation associée au problème d'affectation.

SELECTION qui est une matrice carré de la même taille que **G** avec des **True** et des **False** signalant par un **True** que le zéro dans la matrice **G** aux même ligne est colonne est sélectionné.

MARQUAGE est de même une matrice de **True** et de **False** signalant que le 0 de **G** à la même position est marqué.

COUVERTURE est un tableau associatif avec deux cases :

- La case '**LIG**' est un tableau qui indique par **True** ou **False** si la ligne correspondante est couverte ou non (si **COUVERTURE**['**LIG**'] [2]=**False** alors la ligne 2 (on rapel que la numérotation commence à 0) n'est pas couverte).
- La case '**COL**' est un tableau qui indique par **True** ou **False** si la colonne correspondante est couverte ou non.

On rapel qu'en Python, comme en C ou en Java, les tableaux sont en fait des adresses et que le passage de tableau en paramètre des fonctions ne se fait pas par copie mais par référence. En d'autre terme, toutes modifications d'un tableau dans une fonction entraînent une modification de la même variable dans la fonction mère (fonction principale).

Dans une matrice, on appellera **coordonnée** d'un coefficient, un tableau à double entrée où la première case est l'indice de la ligne du coefficient et la seconde case sa colonne.

Voici une architecture des fonctions qui pourront vous aider à l'écriture de l'algorithme hongrois ¹.

ApparitionOLigne(G) renvoie la matrice où chaque ligne a été décrémenté de son minimum.

ApparitionOColonne(G) renvoie la matrice où chaque colonne a été décrémenté de son minimum.

FinAlgoHongrois(SELECTION) renvoie **True** si l'algorithme est fini (étape 2 de l'algorithme hongrois).

InitMARQUAGE(G) renvoie la matrice **MARQUAGE** initialisée (toutes les cases à **False**).

InitSELECTION(G) renvoie la matrice **SELECTION** initialisée (toutes les cases à **False**).

SectionDes0(G, SELECTION) met à jour la variable **SELECTION** (étape 2 de l'algorithme hongrois).

InitCOUVERTURE(G, SELECTION) renvoie la tableau **COUVERTURE** initialisé (étape 3Initialisation de l'algorithme hongrois).

ZeroNonCouvert(G, COUVERTURE) renvoie les coordonnées d'un zéro non couvert. Si ce n'est pas possible on renvera {0:-1, 1:-1} (étape 3Marquage de l'algorithme hongrois).

marquage(ZERO, MARQUAGE, SELECTION) met à jour la matrice **MARQUAGE** en marquant le zéro situé aux coordonnées passées par le paramètre **ZERO**. Cette fonction renvoie les coordonnées d'un zéro sélectionné sur la même ligne. En cas d'impossibilité la fonction renvoie {0:-1, 1:-1} (étape 3Marquage de l'algorithme hongrois).

numerotation(INIT, SELECTION, MARQUAGE) renvoie une matrice avec la numérotation des 0 (étape 3Initialisation de l'algorithme hongrois).

MajSELECTION(SELECTION, NUMEROTATION) met à jour la variable **SELECTION** en suivant les règles de modifications suivant la **NUMEROTATION** passée en paramètre.

MajG(G, COUVERTURE) ajoute le minimum parmi les coefficient de G non couvert aux lignes couvertes et on le soustrait aux colonnes non couvertes.

AlgoHongrois(G) fait tourner l'algorithme hongrois et renvoie un tableau ou la case i vaut j si i est affecté à j.

1. En particulier, il n'est pas impossible de faire mieux en supprimant des fonctions et/ou en rajoutant d'autre.

Application : Vœux de parcours en S4

A la fin de votre S3 il vous sera demandé de faire des vœux de parcours pour le S4. Au département informatique nous proposons 3 types de parcours :

- Un groupe de **MA** pour math appliquée. Les étudiants seront davantage formés en mathématiques avec deux matières supplémentaires (algèbre II et analyse II).
- Un groupe de **MIG** pour math-info-gestion. Les étudiants seront davantage formés aux métiers de la gestion (chef de projet) avec deux matières en plus (math financières et simulation)
- Deux groupes de **TI** pour technologie de l'informatique avec un approfondissement dans les matières informatiques et une matière supplémentaire (développement mobile).

Pour simplifier le problème nous supposons que chaque groupe est composé d'exactly 25 étudiants et que la promotion validant le S3 pour aller en S4 est de 100 étudiants.

Dans le fichier `VoeuxS4.txt` vous trouverez une version anonyme et partiellement modifiée des vœux émis l'année dernière. Chaque ligne représente un (ancien) étudiant, la première colonne est son identifiant ici rapporté à un simple entier, la seconde colonne la position de vœux pour intégrer le groupe MA, la colonne suivante pour le groupe MIG enfin la dernière pour le groupe TI.

En vous servant du programme que vous avez développé, attribuez à chaque étudiant un groupe. Combien d'étudiant seront dans le groupe de leur dernier choix ? Parmi ceux là, combien avait mi le vœux MIG en premier ?

Pour vous aider, le programme `ExtractVoeux.py` va lire le fichier `VoeuxS4.txt` et construire une variable `VOEUX` étant une matrice à 100 lignes et 4 colonnes correspondant au fichier lu.