

# Homework en R

Neggazi Mohamed Lamine, Raouf Chouik  
Grp 1 ISIL

08 / 02 / 2020

## Introduction :

---

Ce Homework est basé sur la manipulation des graphs avec Language R, le programme script.R va lire une matrice et tester est-ce que cette matrice est une matrice de flux ou non avec des contraintes qu'on a utilisé dans le script comme la dimension de matrice doit être égale ex: ( $\{3,3\}$  ou  $\{6,6\}$ ) et aussi la diagonal = 0 et les éléments selon la diagonale doit être égaux ex: ( $\text{mat}[i,j] == \text{mat}[j,i]$ ).

Après avoir toutes ces conditions on a la composition des cellules qui contiennent les machines ! on va tester si la composition est réalisable ou non avec la valeur donnée donc si la composition n'est pas réalisable le programme (script.R) va afficher un error "pas réalisable!" sinon si elle est réalisable le script va calculer la fonction objectif de cette composition! et il va afficher un graph selon la composition!

[NOTE] SVP!! installer les packages pour que ce script.R génère le Graph Correctement!  
{network, sna, ggplot2 et GGally}

## Objectif :

---

- Vérifier si la matrice donnée est une matrice de flux.
- Tester si la composition est réalisable ou non.
- Calculer la fonction objectif selon la composition donnée.
- Générer le Graph associée selon la composition en colorisant chaque cellule!

## Dépendances :

---

- Version r-base  $\geq 3.5.x$  pour que package "sna" marche correctement!
- Recommandé d'utiliser la dernière version de compilateur c, c++ sous linux :  
`#sudo apt install make build-essential gcc g++ gfortran`
- Optionnelle :  
`#sudo apt install evince zathura mupdf ...`  
Pour l'ouverture de graph (Rplots.pdf)

## Comment Utiliser :

---

L'exécution du script dans la ligne de commande sera comme ceci :

```
#Rscript script.R matrix.txt composition.txt
```

Les fichiers matrix.txt et composition.txt sont déjà fournis avec le script.R pour les tester et modifier à votre objectif!

Matrix.txt est le fichier qui doit contenir la matrice et composition.txt pour la composition des cellules!

[NOTE] On a choisi cette méthode de résolution pour ne pas perdre votre temps à taper toute la matrice et la composition!

## Comment on a réalisé ce Projet (script.R):

---

### Importation des Données:

- On a utilisé la commande "commandArgs" pour ajouter des fichiers dans l'exécution de Rscript dans le terminal pour ne pas taper plusieurs fois!
- On a aussi importé la matrice (matrix.txt) avec la commande read.table pour accéder comme suit: `mat[i, j]` et pour la composition on a choisi `scan` pour lire composition.txt comme une liste et le séparateur est chaque ligne donc le script va lire les lignes comme des strings (type "Character" dans R) et pour sa j'utilise `strsplit` avec espace comme séparateur, donc elle va différencier les éléments dans chaque ligne!
- La commande "paste" pour générer les noms des machines {"M1", "M2", ...}
- La commande "repeat" pour ne pas lire une valeur négative comme valeur de la composition!  
(do {}while ();) en langage C!

### Corps de l'algorithme

- Pour être une matrice de flux il faut 3 contraintes:
  1. La matrice doit être adjacente! (`nbr_ligne == nbr_colons`)
  2. La diagonale de la matrice doit être mise à 0 donc si `i == j` et `mat[i, j] != 0` la matrice n'est pas une matrice de flux!
  3. La matrice doit être symétrique par rapport à la diagonale! `mat[i,j] == mat[j,i]`
- Tester si la valeur donnée avec la composition est réalisable!  
si (la longueur de quelque cellule > valeur donnée) la composition avec valeur n'est pas réalisable!
- Calculer la fonction objectif:  
Pour cela j'ai utilisé la méthode d'élimination des liens des machines qui sont dans la même cellule!  
donc je parcours la liste des cellules en éliminant toutes les machines dans la même cellule!  
Et la somme des liens restants est la  $\{\text{fct objectif} * 2\}$  parce que la diagonale, d'où je divise sur 2!

### Génération du Graph:

Pour la génération du graph j'utilise `ggnet2` comme ça:

- Fonction "network" pour créer un réseau des nœuds qui sont connectés avec des liens entre eux! le type de matrice est "adjacent" et pas de cible (flèche) pour les liens et les autres arguments pour établir les valeurs des liens! "weights of edges"
- "option" est pour différencier entre les cellules!

- “myColor” pour définir mes couleurs pour les différentes cellules!
- La fonction `ggnet2()` est pour générer le Graph Associer avec mes paramètres!

À la fin j'utilise un truc pour ouvrir le graph généré avec `ggnet2()` en supposant que `evince` est installé (par défaut il est installé sous ubuntu) Je suis désolé si ça ne marche pas! pour cela vous devez manuellement l'ouvrir!

[NOTE] J'ai ajouté un fichier nommé `sample.pdf` si vous avez des problèmes avec l'installation des packages! pour le reste ça marchera très bien!!