


Nom : Prénom :	<b>THEME 3</b> <b>Les réseaux sociaux</b>	RS-TP2 1/5	
Date :	<b>Les graphes</b>		

### L'expérience de Milgram (1967) :

Milgram envoie 60 lettres à des recrues de la ville d'Omaha dans le Nebraska. Il leur demande de faire suivre cette lettre à un agent de change, vivant à une adresse fournie, dans la ville de Sharon dans le Massachusetts. Les participants pouvaient seulement passer les lettres, de main à main, à des connaissances personnelles qu'ils pensaient être capables d'atteindre l'objectif, directement ou via les amis des amis. Le célèbre article de 1967 de Milgram décrit le fait qu'une lettre ne mit que quatre jours pour atteindre sa destination, mais négligea de mentionner que seulement 5 % des lettres réussirent à rejoindre leur cible. On peut en conclure que nous ne sommes jamais très éloignés les uns des autres...

[https://www.youtube.com/watch?time\\_continue=41&v=P7xltpdmXoc&feature=emb\\_logo](https://www.youtube.com/watch?time_continue=41&v=P7xltpdmXoc&feature=emb_logo)



Combien d'intermédiaires sont utilisés pour porter la lettre à son destinataire ?  
Combien de lettres sont arrivés à destination ?

Réponse : .....  
Réponse : .....

Voici un graphe qui modélise des liens entre des personnes.




Comment interpréteriez-vous ce graphe ?

.....

.....

.....

.....

Nom : Prénom :	<b>THEME 3</b> Les réseaux sociaux	RS-TP2 2/5	
Date :	<b>Les graphes</b>		

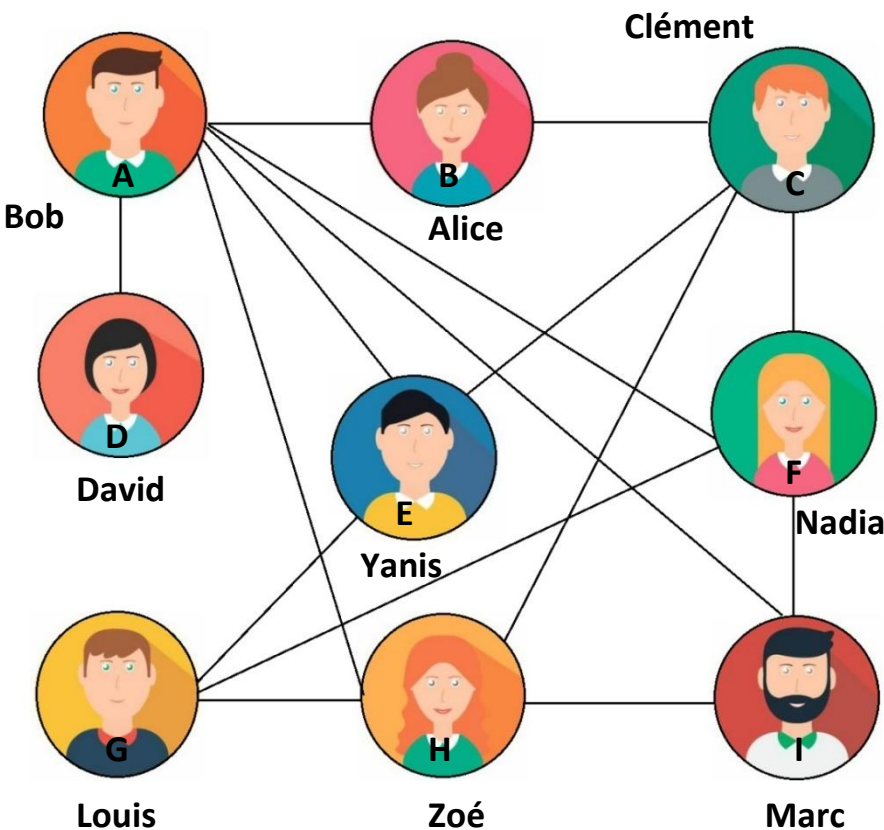
## Quelques définitions sur les graphes

Un graphe est constitué de sommets (nœuds) et d'arêtes (segments reliant deux sommets).

- **Chaîne** : Dans un graphe, une chaîne reliant un sommet x à un sommet y est définie par une suite finie d'arêtes consécutives, reliant x à y.
- **Distance entre 2 sommets** : La distance entre deux sommets d'un graphe est le nombre minimum d'arêtes d'une chaîne allant de l'un à l'autre.
- **Écartement** : L'écartement d'un sommet est la distance maximale existant entre ce sommet et les autres sommets du graphe.
- **Centre** : On appelle centre d'un graphe, le sommet d'écartement minimal. (le centre n'est pas nécessairement unique).
- **Rayon** : On appelle rayon d'un graphe, l'écartement minimal d'un centre du graphe.
- **Diamètre** : On appelle diamètre d'un graphe, la distance maximale entre deux sommets du graphe.


## Application

Avec les précédentes définitions déterminer le(s) centre(s), le rayon et le diamètre du graphe ci-dessous.



Exemples
- A-B-C-F-I-H-G est une chaîne.
- D-A-B-C-E est une autre chaîne
(Il y en a plusieurs dans un graphe)
- La distance entre le sommet A et le sommet G de la chaîne A-E-G est de 2.
- La distance entre le sommet C et le sommet D est de 3
- L'écartement du sommet B est de 2
- L'écartement du sommet C est de 3



Nom : Prénom :	THEME 3 Les réseaux sociaux	RS-TP2 4/5	
Date :	Les graphes		

**Programmation de la matrice d'adjacence en langage PYTHON**

Ouvrir l'IDLE PYTHON, Pyscript ou bien EduPython (Pyscript inclus)

Saisir le code suivant :

```

Fichier  Edition  Rechercher  Affichage  Projet  Exécuter  Outils  Aide
Nouveau Fichier...
amis=[[0,1,0,1,0,1,1,0,1,1],
[1,0,1,0,0,0,0,0,0],
[0,1,0,0,0,1,1,1,0],
[1,0,0,0,0,0,0,0,0],
[1,0,1,0,0,0,1,0,0],
[1,0,1,0,0,0,1,0,1],
[0,0,0,0,1,1,0,1,0],
[1,0,1,0,0,0,1,0,1],
[1,0,0,0,0,1,0,1,0]]
10
Nom=["Bob","Alice","Clément","David","Yanis","Nadia","Louis"," Zoé","Marc"]
for i in range(0,9,1):
    nombamis=0
    for j in range(0,9,1):
        if amis[i][j]==1:
            nombamis=nombamis+1
    print(Nom[i]," a ",nombamis, "amis dans le réseau social")

```

Ecrivez le résultat obtenu dans le cadre ci-dessous

.....

.....


.....

.....

.....

.....

.....

Nom : Prénom :	<b>THEME 3</b> <b>Les réseaux sociaux</b>	RS-TP2 5/5	
Date :	<b>Les graphes</b>		

## Réalisation d'un graphique du réseau social

Ouvrir l'IDLE PYTHON ou Pyscript Saisir le code suivant

```

Fichier  Edition  Recherche  Affichage  Projet  Exécuter  Outils  Aide
nouveau Fichier...
""" l'importation préalable de la bibliothèque networkx s'obtient en exécutant la commande conda install networkx dans la fenêtre shells"""
import networkx as nx          # importation de la bibliothèque permettant la création de graphes
import matplotlib.pyplot as plt # importation de la bibliothèque permettant les tracés

amis = [[0,1,0,1,1,1,0,1,1], # Matrice d'adjacence
        [1,0,1,0,0,0,0,0,0],
        [0,1,0,0,0,1,1,1,0],
        [1,0,0,0,0,0,0,0,0],
        [1,0,1,0,0,0,1,0,0],
        [1,0,1,0,0,0,1,0,1],
        [0,0,0,0,1,1,0,1,0],
        [1,0,1,0,0,0,1,0,1],
        [1,0,0,0,1,0,1,0]]

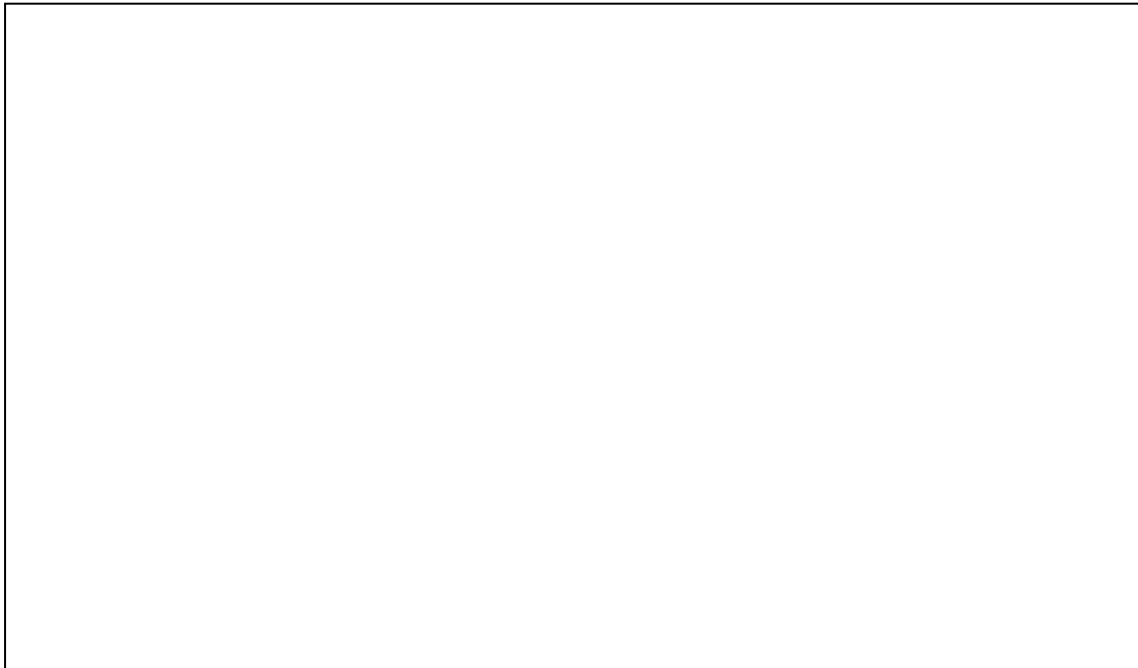
arcs=[] # création de la liste contenant les arcs
for l in range(len(amis[0])): # construction de la liste des arcs à partir de la matrice d'adjacence
    for c in range(len(amis[0])):
        if amis[l][c]==1:
            arcs.append([l,c])

noms={0:'Bob',1:'Alice',2:'Clément',3:'David',4:'Yanis',5:'Nadia',6:'Louis',7:'Zoé',8:'Marc'} # noms apparaissant dans les noeuds
G=nx.Graph() # création d'un graphe vide nommé G
G.add_nodes_from([0,1,2,3,4,5,6,7,8]) # création des noeuds du graphe G
G.add_edges_from(arcs) # création des arcs du graphe G
nx.draw_networkx(G,labels=noms, node_size=3000) # fabrication du graphe G avec noms et taille des noeuds

plt.show() # tracé du graphe G

```

Représentez le graphe obtenu après avoir exécuté le programme



Conclusion : .....

.....

.....