



SAE 2.02 - Rapport

Sommaire

1/ Travail effectué

2/ Répartition des tâches

3/ Réponses

4/ Mesures expérimentales

5/ Conclusion

Travail effectué

Le travail à réaliser consiste à implémenter un programme en Python pour répondre à une série de requêtes permettant d'évaluer la centralité des acteurs à Hollywood. Le programme utilisera la bibliothèque NetworkX pour représenter le graphe des collaborations entre les différents acteurs.

Différentes requêtes ont donc été implémentées au cours de cette SAE :

```
# Q1
> def json_vers_nx(chemin): ...

# Q2
> def collaborateurs_communs(G, u, v): ...

# Q3
> def collaborateurs_proches(G, u, k): ...

> def est_proche(G, u, v, k=1): ...

> def distance_naive(G, u, v): ...

> def distance(G, u, v): ...

# Q4
> def centralite(G, u): ...

> def centre_hollywood(G): ...

# Q5
def eloignement_max(G):
>     """ ...
    return nx.diameter(G)

# Bonus
> def centralite_groupe(G, S): ...
```

Répartition des tâches

Yassine FELLAH

```
def json_vers_nx(chemin);  
def collaborateurs_communs(G,u,v);  
def collaborateurs_proches(G,u,k);  
def est_proche(G,u,v,k=1);  
def distance(G,u,v);
```

Alexys MARIE

```
def distance_naive(G,u,v);  
def centralite(G,u);  
def centre_hollywood(G);  
def eloignement_max(G:nx.Graph);  
def centralite_groupe(G,S);
```

Fait à 2

Fichier oracle.py, tests des fonctions
implémentées

Réponses

6.2/

En théorie des graphes, les collaborateurs en commun entre deux acteurs sont les nœuds voisins communs des deux nœuds représentant les acteurs. La borne inférieure sur le temps d'exécution de cette fonction est $O(n)$, avec n nombre total de nœuds dans le graphes.

6.3/

En utilisant la fonction `collaborateurs_proches`, nous pouvons déterminer si un acteur se trouve à distance k d'un autre acteur en effectuant un parcours en largeur BFS de l'acteur initial et en vérifiant si l'acteur final est atteint dans k étapes. La complexité de cette méthode est $O(n+m)$, où n est le nombre de nœuds et m est le nombre d'arêtes dans le graphe.

6.4/

Le script vérifie et imprime si tous les acteurs sont à une distance de 6 ou moins les uns des autres pour chaque fichier de données.

Mesures experimentales

Les fonctions et le graphe s'exécutent sans problème.

```
Type: Graph
Number of nodes: 1461
Number of edges: 17861
Average degree: 24.4504
```

```
-----
Collaborateurs commun avec Al Pacino et Robert de Niro : {'Bette Midler', 'Sebastian Maniscalco', 'Leopoldo Trieste', 'Rob Dyrdek', 'Frank John Hughes', 'Bobby Cannavale', 'Abe Vigoda', 'Woody Allen', 'Louis Cancelmi', 'Stephen Graham', 'D
```

```
Robert de Niro et Al Pacino sont proches à distance = 2 ? True
```

```
Distance naïve entre Al Pacino et Robert De Niro : 1
```

```
Distance entre Al Pacino et Robert De Niro : 1
```

```
Centralité de Al Pacino : 2
```

```
L'acteur le plus central de Hollywood : Al Pacino
```

```
Éloignement maximal dans le graphe : 3
```

```
Centralité du groupe d'Al Pacino et Robert de Niro : 2510.5
```

```
Collaborateurs proches de Al Pacino : {'Louis Cancelmi', 'Eileen Brennan', 'Mark E', 'J. P. Bumstead', 'Jeremy Irons', 'Dyan', 'Lynne Thigpen', 'Tura Nakamura', 'iven', 'Glen Powell', 'Margaret Qualley', '(actor)|William Marshall', 'Darren Golds', 'Allan Rich', 'Keith Prentice', 'Caroll', 'Paul Dooley', 'Shields and Yarnell|Roberttieri', 'Nicholas Hammond', 'Bob Dishy', 'y Williams', 'Jeff Bridges', 'Sterling K. rovan O'Donnell', 'Robert Guillaume', 'Dre', 'Lance Reddick', 'Ed Bakey', 'Phil Brown (act', 'Sarah Douglas (actress)|Sarah Douglas', 'Benjami', 'Rebecca Rittenhouse', 'Anne Heche', 'rdson', 'Bridget Moynahan', 'Tony Roberts', 'Billy Porter (actor)|Billy Porter', 'B
```

