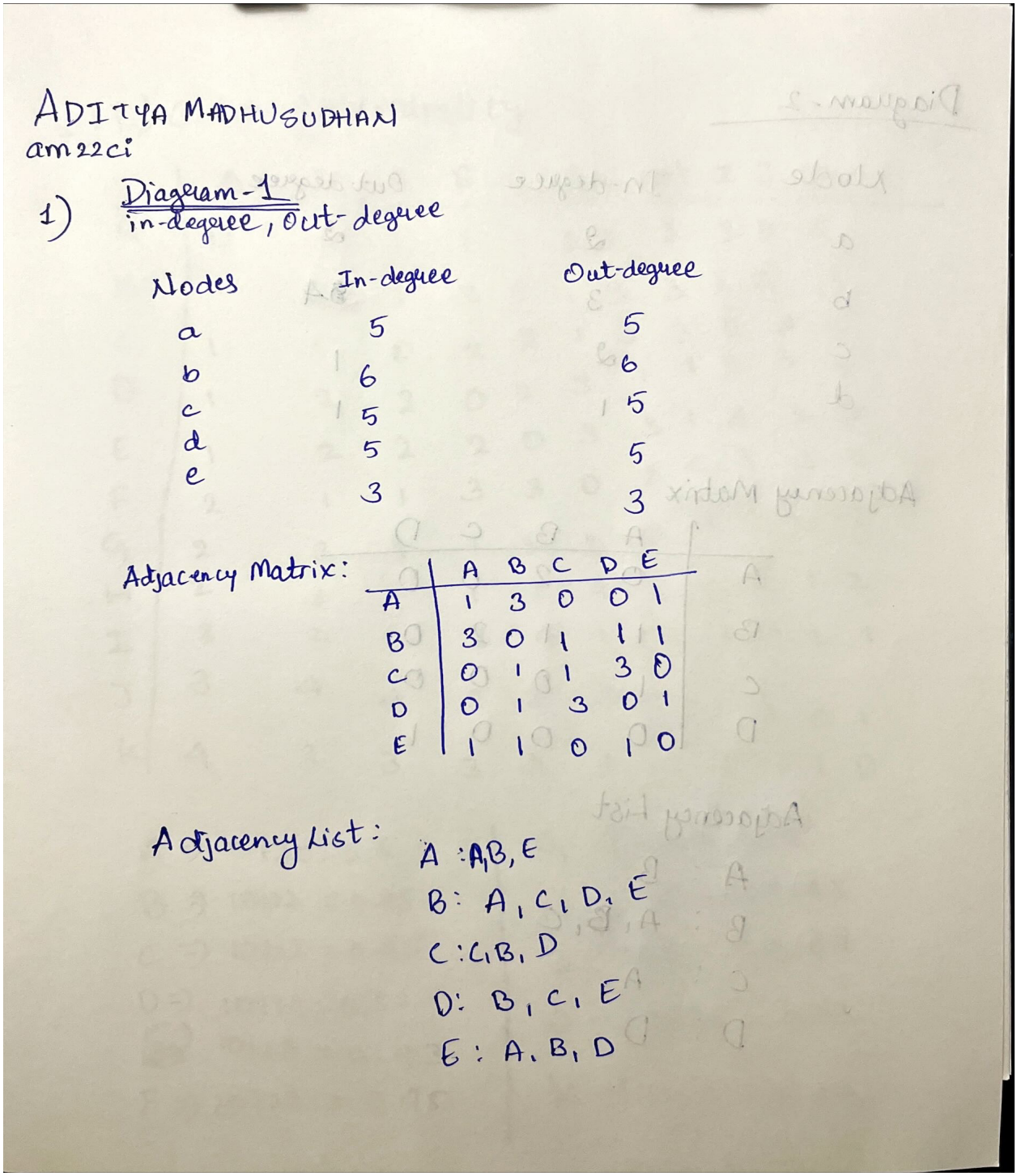
IDS Assignment 2

Problem 1



ADITYA MADHUSUDHAN
am22ci

1) Diagram-1
in-degree, Out-degree

Nodes	In-degree	Out-degree
a	5	5
b	6	6
c	5	5
d	5	5
e	3	3

Adjacency Matrix:

	A	B	C	D	E
A	1	3	0	0	1
B	3	0	1	1	1
C	0	1	1	3	0
D	0	1	3	0	1
E	1	1	0	1	0

Adjacency List:

A : A, B, E
B : A, C, D, E
C : C, B, D
D : B, C, E
E : A, B, D

Diagram - 2

Node	In-degree	Out-degree
a	2	2
b	3	4
c	2	1
d	1	1

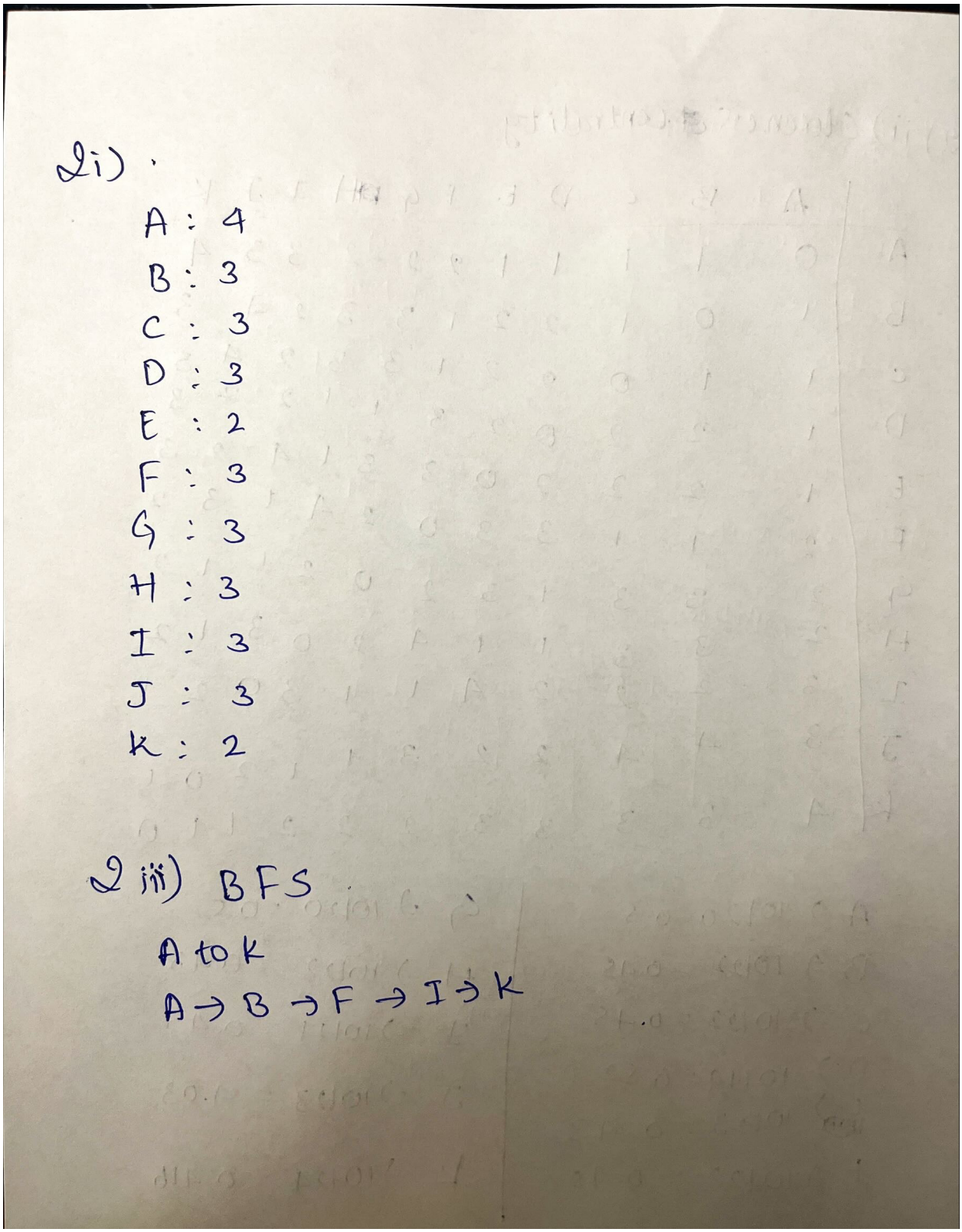
Adjacency Matrix

	A	B	C	D
A	0	2	0	0
B	1	1	2	0
C	1	0	0	0
D	0	0	0	1

Adjacency List

A : B
B : A, B, C
C : A
D : D

Problem 2



2i).

A : 4
B : 3
C : 3
D : 3
E : 2
F : 3
G : 3
H : 3
I : 3
J : 3
K : 2

2 iii) BFS

A to K

A → B → F → I → K

2) ii) Closeness centrality

	A	B	C	D	E	F	G	H	I	J	K
A	0	1	1	1	1	2	2	2	3	3	4
B	1	0	1	2	2	1	3	3	2	4	3
C	1	1	0	2	2	1	3	3	2	4	3
D	1	2	2	0	2	3	1	1	2	2	3
E	1	2	2	2	0	3	3	1	4	2	3
F	2	1	1	3	3	0	2	4	1	3	2
G	2	3	3	1	3	2	0	2	1	1	2
H	2	3	3	1	1	4	2	0	3	1	2
I	3	2	2	2	4	1	1	3	0	2	1
J	3	4	4	2	2	3	1	1	2	0	1
K	4	3	3	3	3	2	2	2	1	1	0

A ⇒ 10/20 = 0.5

B ⇒ 10/22 = 0.45

C ⇒ 10/22 = 0.45

D ⇒ 10/19 = 0.52

E ⇒ 10/23 = 0.43

F ⇒ 10/22 = 0.45

G ⇒ 10/20 = 0.5

H ⇒ 10/22 = 0.48

I ⇒ 10/21 = 0.47

J ⇒ 10/23 = 4.03

K ⇒ 10/24 = 0.416

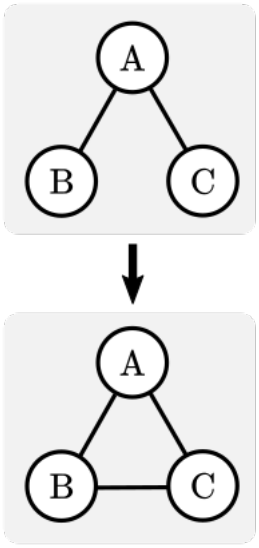
Problem 3

Jupyter Notebook code file name (3.ipynb)

I. By exploring the different graph representations in the networkx the random layout did not give a representation of what the data in graph is and the spiral is more useless, the above dataset representation the spring layout visualization looks useful. The nodes in the graph looks highly interconnected as expected from a relation ship based design, as facebook creates a node for each user and connects the friends of each user to the respective node. This means we might have clusters in the graph dataset, as people of a group might be separated like for example there are three instances in the above graph.

II. The graph has edges 88234 and nodes 4039, the average degree of node is 43.69101262688784. By the plot we can see that the graph has highest number of 1 path between the nodes in the graph making a different meaning as they might be new to the set or they outliers. Also the degree centrality that show the number of links held by each node showed that the users have centrality lesser than 0.05. This also seems that they are not very interconnected well in this network so many nodes have low degree centralities. The graph also represents the nodes with high degree centralities. By the observation of the centrality only we can say that the dataset dose not follow power law. The power distribution for graph is checked using the https://pypi.org/project/powerlaw/ library. The power law distribution for the graphs degree distribution is false as the power is lesser than 0.05, the power is 1.0109665086696757e-08 is very small to follow the same.

III. From the coefficient histogram we can see the highest count of the nodes reside towards the 1.0 and this might be possible due to the high number of nodes available in the network. Also the most nodes lie just off from 0.4 to 0.7 and a bit upwards probably according the visual information. Also the new nodes addition to the network can form between the nodes as a node will be linked to a friend. for example, a new user will link himself/herself to a friend on the network so the new node will be place between a existing link.



The network also acts as a triadic line the property of three nodes being related in the future. This also makes the position of mutual friends a relation between the nodes and the trust node. The above explains the transitive property the triadic closure that occur, if A and B are friends, A and C are friends, this means that B and C are

mutual to A. This link might result in 2 ways a new prospective friend or a stranger mutual to B and C. This network behaviors can lead to structural and informational positions. The structural construction arises from the propensity toward high cluster ability. The informational construction comes from the assumption that an individual knows something about a friend's friend, as opposed to a random stranger.

Problem 4

Jupyter Notebook code file name (4.ipynb)

i) TF IDF of the document is = 0.5217391304347826

ii)

```
[[1.         0.07190658 0.67881986 0.37747408 0.         0.72207718]
 [0.07190658 1.         0.         0.15118579 0.98994949 0.13158324]
 [0.67881986 0.         1.         0.18142875 0.         0.3656746 ]
 [0.37747408 0.15118579 0.18142875 1.         0.08908708 0.50170051]
 [0.         0.98994949 0.         0.08908708 1.         0.        ]
 [0.72207718 0.13158324 0.3656746  0.50170051 0.         1.        ]]
```

iii) The TF-IDF is not a good way to handle multi-dimensional data due to its inability to weight the feature, this problem might lead to biased results. To take an example text corpus has multiple attributes such as name, address, phone, date, and more. This case is not handled properly in the traditional TF-IDF.
To overcome this problem, we can use the weighting of attributes, where according to problem statement we can weight the attributes and improve the results. One other approach is the LDA approach that uses complex relations between he attributes to determine the corpus.

Problem 5

Jupyter Notebook code file name (5.ipynb)

i) Done in code

ii) Stemming and lemmatization are techniques used to reduce words to their base or root form. Stemming is a process that stems or removes last few characters from a word, often leading to incorrect meanings and spelling. Lemmatization considers the context and converts the word to its meaningful base form, which is called Lemma. Both stemming and lemmatization is implement on the preprocessed data. The stemmed and lemmatized data are stored in separate variables. Stemming is a faster and simpler technique compared to lemmatization, as it involves only removing the suffix from a word. However, stemming may not always result in a valid word, as the base form may not exist in the English language. On the other hand, lemmatization maps a word to its base form based on a dictionary, which ensures that the resulting word is valid. Lemmatization is a slower technique compared to stemming. Also, lemmatization is preferred over stemming as it results in more meaningful base forms. However, in situations where speed is a crucial factor, stemming may be preferred.

iii) NLTK Gutenberg dataset consists of 18 classic English texts from Gutenberg Project. Some potential biases and limitations:

We have to consider these limitations and potential biases with the Gutenberg dataset.

a. Though the dataset has a wide range of genres, it is still a relatively small dataset compared to other modern text datasets, which can limit the generalizability of models trained on it.
b. The Gutenberg dataset texts are from 19th century or earlier, mostly the data set will reflect the references of cultural and history of that time
c. The Gutenberg dataset only includes texts written in English, which means that it may not be representative of other languages or cultures.
Improve the limitations.
a. First step might be to generalizability of the LDA model, use larger datasets.
b. Also we can combine multiple datasets to increase the diversity of the training data.
c. Use additional visualization techniques, such as word clouds or topic coherence measures, to help users understand the meaning of the topics more easily.
d. Hyperparameters should be optimized, we can use techniques such as Bayesian optimization to find optimal hyperparameters.

Problem 6

Jupyter Notebook code file name (6.ipynb)

i)

```
In [79]: in_degree.argmax()
Out[79]: 56

In [80]: in_degree.index[in_degree.argmax()]
Out[80]: 'Mohit Kulkarni'

In [81]: in_degree[in_degree.argmax()]
Out[81]: 18.0

In [82]: out_degree.argmax()
Out[82]: 32

In [83]: out_degree.index[out_degree.argmax()]
Out[83]: 'G Shanmukh Vishnu'

In [84]: out_degree[out_degree.argmax()]
Out[84]: 34.0
```

ii) Degree centrality

In [87]:
```
# Degree Centrality
nx.degree_centrality(adjacent_matrix_graph)
```

Out[87]:
```
{'AIdan Mahoney': 0.03529411764705882,
 'Abhiram Vissamsetty ': 0.1764705882352941,
 'Abhishek Pavshe': 0.0,
 'Abhishek Prakash': 0.16470588235294117,
 'Aditya Madhusudhan': 0.3058823529411765,
 'Aditya Sugandhi': 0.23529411764705882,
 'Agamjot singh': 0.11764705882352941,
 'AkhilRaj Tirumalasetty ': 0.2823529411764706,
 'Amar Adilovic': 0.07058823529411765,
 'Andrew Franklin': 0.023529411764705882,
 'Andrew Piaget': 0.0,
 'Anusha Dayanand': 0.27058823529411763,
 'Aojie Yang': 0.08235294117647059,
 'Archit Khandelwal': 0.3176470588235294,
 'Austin Miller': 0.058823529411764705,
 'Ayoola Oguntoyinbo': 0.047058823529411764,
 'Befekadu Lakew': 0.0,
 'Benjamin Friedman ': 0.07058823529411765,
 'Brady Henderson': 0.08235294117647059,
 'Caitlyn Jessee': 0.047058823529411764,
 'Calvin Smyk': 0.07058823529411765,
 'Candies Woods': 0.03529411764705882,
 'Charishma Kuna': 0.08235294117647059,
 'Chris Pierre Paul': 0.03529411764705882,
 'Cole McGuire': 0.08235294117647059,
 'Damarco Rutlin': 0.0,
 'Davone Simmons': 0.058823529411764705,
 'Domen Demsar': 0.0,
 'Esha Nandy': 0.16470588235294117,
 'Faraz Ahmad': 0.023529411764705882,
 'Fazlidddin Mirsoatov': 0.058823529411764705,
 'Frecks Bertrand': 0.0,
 'G Shanmukh Vishnu': 0.4117647058823529,
 'Hannah Housand': 0.08235294117647059,
 'Harshitha Kamineni': 0.2,
 'Holly Jordan': 0.011764705882352941,
 'Hyunjin Yi': 0.07058823529411765,
 'Ian Cleaver': 0.011764705882352941,
 'Ibrahim Afridi': 0.058823529411764705,
 'Jack Moran': 0.011764705882352941,
 'Jacob Guthrie': 0.0,
 'Jacquelyn Nogueras': 0.07058823529411765,
 'Jason St. John': 0.08235294117647059,
 'Juan R Reza': 0.07058823529411765,
 'Keerthi Reddy Vudem': 0.21176470588235294,
 'Kirsten Blair': 0.058823529411764705,
 'Linwei Jiang': 0.047058823529411764,
 'Logan Poland': 0.047058823529411764,
 'Lucas Zavalia': 0.047058823529411764,
 'Marcelo Paesani': 0.03529411764705882,
 'Margaret Rivas': 0.07058823529411765,
 'Marija Travoric': 0.058823529411764705,
 'Mason Ballard': 0.011764705882352941,
 'Mauricio Espinoza': 0.023529411764705882,
 'Md. Masum Al Masba': 0.03529411764705882,
 'Mike Zahorec': 0.10588235294117647,
 'Mohit Kulkarni': 0.24705882352941178,
 'Nikhila Vudem': 0.21176470588235294,
 'Nikola Vuckovic': 0.10588235294117647,
 'Pooja Mhetre': 0.21176470588235294,
 'Preston Turnage': 0.058823529411764705,
 'Reece Gabbett': 0.0,
 'Rodjina Pierre Louis': 0.0,
 'Ross Kane': 0.047058823529411764,
 'Ryan Fontaine': 0.07058823529411765,
 'Sabrina Callejo': 0.058823529411764705,
 'Sai Joshitha Chitikala': 0.1764705882352941,
 'Sai Kalyan Tarun Tiruchirapally': 0.2588235294117647,
 'Sai madhavi gujju ': 0.11764705882352941,
 'Shaoying Wang': 0.047058823529411764,
 'Shaurya Tiwari': 0.23529411764705882,
 'Sophia Villalonga': 0.08235294117647059,
 'Srikanth Mallipeddi': 0.24705882352941178,
 'Teja Potu': 0.16470588235294117,
 'Tejas Vedagiri': 0.29411764705882354,
 'Varun Totakura': 0.27058823529411763,
 'Venkata Sivasai Phani Praveen': 0.32941176470588235,
 'Vishwam Shah': 0.23529411764705882,
 'Wen Huang': 0.058823529411764705,
 'Xiaoxia Che': 0.07058823529411765,
 'YIJIA Wen': 0.07058823529411765,
 'Yasaswi Akash Gunda': 0.15294117647058825,
 'Yunzheng Lyu': 0.08235294117647059,
 'Zhixi Lin': 0.07058823529411765,
 'puneeth reddy motukuru Damodar': 0.1411764705882353,
 'sparsh khare': 0.15294117647058825}
```

iii) closeness centrality

In [88]:
```
nx.closeness_centrality(adjacent_matrix_graph)
```

Out[88]:
```
{'AIdan Mahoney': 0.1849384001388166,
 'Abhiram Vissamsetty ': 0.30141402714932125,
 'Abhishek Pavshe': 0.0,
 'Abhishek Prakash': 0.30434037692747,
 'Aditya Madhusudhan': 0.3463763405914852,
 'Aditya Sugandhi': 0.3150458173219036,
 'Agamjot singh': 0.2836837902581847,
 'AkhilRaj Tirumalasetty ': 0.31824425201552703,
 'Amar Adilovic': 0.27258312020460357,
 'Andrew Franklin': 0.21180445151033386,
 'Andrew Piaget': 0.0,
 'Anusha Dayanand': 0.31824425201552703,
 'Aojie Yang': 0.17760373271121477,
 'Archit Khandelwal': 0.33526266121421827,
 'Austin Miller': 0.20691127936323042,
 'Ayoola Oguntoyinbo': 0.2737734395068071,
 'Befekadu Lakew': 0.0,
 'Benjamin Friedman ': 0.3215082956259427,
 'Brady Henderson': 0.243000455996352,
 'Caitlyn Jessee': 0.023529411764705882,
 'Calvin Smyk': 0.2348094293897334,
 'Candies Woods': 0.2082861051397303,
 'Charishma Kuna': 0.2737734395068071,
 'Chris Pierre Paul': 0.24020734730673876,
 'Cole McGuire': 0.2690734662963898,
 'Damarco Rutlin': 0.0,
 'Davone Simmons': 0.20691127936323042,
 'Domen Demsar': 0.0,
 'Esha Nandy': 0.2902505446623094,
 'Faraz Ahmad': 0.2702332657200811,
 'Fazlidddin Mirsoatov': 0.278640522875817,
 'Frecks Bertrand': 0.0,
 'G Shanmukh Vishnu': 0.38462648863226273,
 'Hannah Housand': 0.31824425201552703,
 'Harshitha Kamineni': 0.30732410611303346,
 'Holly Jordan': 0.1722365869424693,
 'Hyunjin Yi': 0.2478028365496396,
 'Ian Cleaver': 0.17660314830157414,
 'Ibrahim Afridi': 0.23838067546410197,
 'Jack Moran': 0.26678347934918645,
 'Jacob Guthrie': 0.0,
 'Jacquelyn Nogueras': 0.26453214197071234,
 'Jason St. John': 0.243000455996352,
 'Juan R Reza': 0.2623184838789072,
 'Keerthi Reddy Vudem': 0.31036691904484565,
 'Kirsten Blair': 0.20691127936323042,
 'Linwei Jiang': 0.18119687181230873,
 'Logan Poland': 0.023529411764705882,
 'Lucas Zavalia': 0.023529411764705882,
 'Marcelo Paesani': 0.2082861051397303,
 'Margaret Rivas': 0.27740760020822486,
 'Marija Travoric': 0.2478028365496396,
 'Mason Ballard': 0.1722365869424693,
 'Mauricio Espinoza': 0.24020734730673876,
 'Md. Masum Al Masba': 0.24585928489042672,
 'Mike Zahorec': 0.29433858050262357,
 'Mohit Kulkarni': 0.3282414536495226,
 'Nikhila Vudem': 0.30732410611303346,
 'Nikola Vuckovic': 0.2478028365496396,
 'Pooja Mhetre': 0.30732410611303346,
 'Preston Turnage': 0.2339332748024583,
 'Reece Gabbett': 0.0,
 'Rodjina Pierre Louis': 0.0,
 'Ross Kane': 0.2215339846185824,
 'Ryan Fontaine': 0.22964878258995905,
 'Sabrina Callejo': 0.26014156700024405,
 'Sai Joshitha Chitikala': 0.30141402714932125,
 'Sai Kalyan Tarun Tiruchirapally': 0.3166369578134284,
 'Sai madhavi gujju ': 0.29854341736694673,
 'Shaoying Wang': 0.15328635121530274,
 'Shaurya Tiwari': 0.3150458173219036,
 'Sophia Villalonga': 0.21470588235294116,
 'Srikanth Mallipeddi': 0.31036691904484565,
 'Teja Potu': 0.30434037692747,
 'Tejas Vedagiri': 0.3444731738849386,
 'Varun Totakura': 0.3370651486401012,
 'Venkata Sivasai Phani Praveen': 0.3231655548817465,
 'Vishwam Shah': 0.3150458173219036,
 'Wen Huang': 0.1838537174400552,
 'Xiaoxia Che': 0.15328635121530274,
 'YIJIA Wen': 0.15328635121530274,
 'Yasaswi Akash Gunda': 0.29433858050262357,
 'Yunzheng Lyu': 0.16808074436208797,
 'Zhixi Lin': 0.2239075630252101,
 'puneeth reddy motukuru Damodar': 0.2971285196543072,
 'sparsh khare': 0.30434037692747}
```

Problem 7

Jupyter Notebook code file name (7.ipynb)

1.

	Unnamed: 0	Very interested	Somewhat interested	Not interested
0	Big Data (Spark / Hadoop)	1332	729	127
1	Data Analysis / Statistics	1688	444	60
2	Data Journalism	429	1081	610
3	Data Visualization	1340	734	102
4	Deep Learning	1263	770	136
5	Machine Learning	1629	477	74

2. Heuristics of the data