PART-C

11. To Combine the advantages of matrices and node link diagrams, we use "Matrix Explorer."

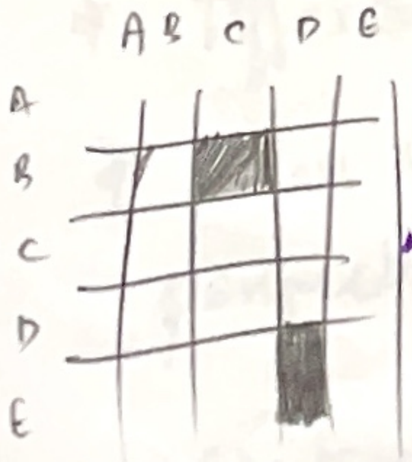
The main goal is to visualize the network.

It Consists of the following steps:

1. Initiation of Exploration
2. Explore interactively and iteratively
3. Finding a consensus in data
4. Present findings

Initiation of Exploration:

* Matrix representation is readably easy to create and has faster initiation for exploration because the rendering time is low
* In matrix representation, the black colored cells imply that there is communication b/w the row and column.



Matrix

* In traditional forced layout, as the graph grows the connection b/w the nodes cross, there is overlapping in the nodes and it is not visually readable
* Dense networks are present and peripheral nodes might have less connections!

2.2/P13

DAR
G

SSN 4

* In matrices, this readability isn't an issue
* In dense networks, then many black cells are present. Gray cells represent that there are no connection between the nodes.
* If properly reordered it could help us find many insightful details
* A particular row or column in black cells implies there is a match from an institution / organization.

Iterative Exploration:

* The exploration process in itself is iterative.
* To make it interative, we manipulate the visualizations.
* Both node-link and matrix representation is allowed.
* It includes the following for interactive

* Iteratively find the visual characteristics:
  characteristics like color, label, size etc.
* Iterative layout and order:
  Manually specifying where the nodes position must be
* Automated layout and order:
  Using computer algorithms to specify layout and order

* Computer assisted la
  subset of the net
* Filtering: Filtering or
  a text f
* Combining: Data
  similar
* Overview + Focus: R
  u
  Focus content: Use bi
  fisheye for
  Tree map: Used for
  Filtering is used to f
  Find a Consensus in Da

Find consensus in Dat

* Different visualiz
* When reordering, l
  perspective / clust
* Solution is to use
  that are prese
* If a node's mem
  differentiate them by

* Computer assisted layout and ordering: only a subset of the network uses layout and ordering algorithms to

* Filtering: Filtering or removal of elements based on value typed in a text field or from a list.

* Combining: Doing / aggregating the elements that are similar

* Overview + Focus: Reordering matrices, treemaps, flow etc are used:

↳ Focus Content: Use birds eye for overview of network and navigation, use fisheye for magnifying details.

↳ Tree map: Used for macrostructure & micro

↳ Filtering is used to find communities within the data.

~~Finding Consensus in Data~~:

Finding consensus in Data:

* Different visualizations provide with insights in data.

* When reordering, each time matrices provide a different perspective / cluster

* Solution is to iteratively do reordering, and check clusters that are present in almost all representations.

* If a node's membership is doubtful, then could differentiate them by using lighter colours.

SSN

Present Findings:

* Matrix representation is helpful if we want overview of the entire network
* Node link is useful when we have to present to a wider audience.
* Multiple views of ~~node link~~ node-link representations could be used.
* Matrix Explorer provides with features to take pictures ~~as well~~ of the views as well.

2.

```
import networkx as nx
import matplotlib.pyplot as pt

G = nx.generators.small.krackhardt_kite_graph(7)
#a. Display adjacency list
print(G.adj)
#b. Print neighbors of each node
for node in G.nodes():
    print("The neighbours of", node, "are")
    print(G[node])

#c. Depth first search with node 0 as start point
import networkx.algorithms.traversal
def dfs_find(G, start, visited=[], edges=[]):
    visited.append(start)
    print(start)
    for neighbour in G[start]:
        if neibbour not in visited:
```

```
        edges.appd(
        dfs_find(G, ne
    return edges
dFs(G, 0)

#d. Shortest path from
print(nx.algorithms.
#e. Average shortest pa
print(nx.algorithms.
#
#
#
```

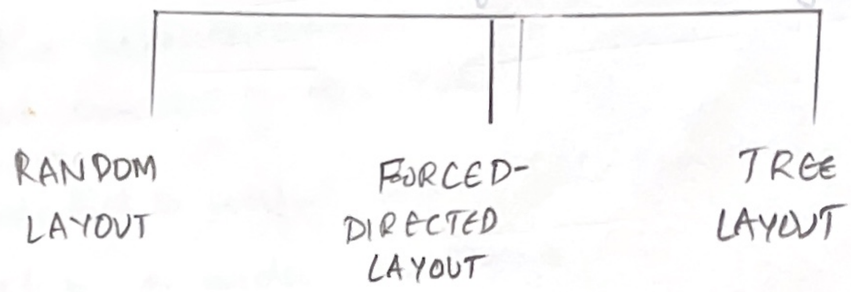
```
        edges.append((start, neighbor))
        dfs_find(G, neighbor, visited, edges)
    return edges
dfs(G, 0)

# d. shortest path from node 0 to 7
print(nx.algorithms.shortest_path(G, 0, 7))

# e. Average shortest path
print(nx.algorithms.average_shortest_path(G))

# Depth first search
# print(dict(nx.algorithms.dfs_successors(G, 0)))
# print
```

PTO for Part B & A

SSN

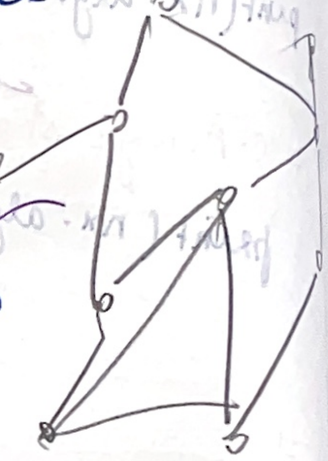
PART-B :

1.



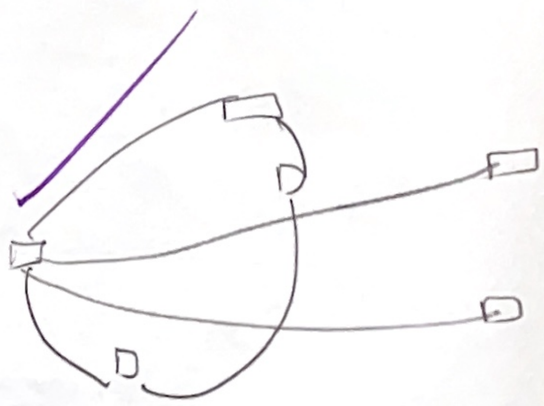
Random Layout :

* Nodes are present in random geographical positions.
* Links are also ~~in~~ ~~can~~ randomized
* Cannot be used for more than 1000 nodes
* The time for drawing the network is linear with number of nodes $\approx O(n)$
* Can be used for large networks



Fixed-directed Layout

* Nodes are like repelling objects and the edges are like springs
* They follow Hookes and Coulomb's law
* Initially it starts as a random layout



* The nodes then
* This proces
  How nod
* It make
* It take
  no

User centric Visualization
* Here th

Tr

Q.

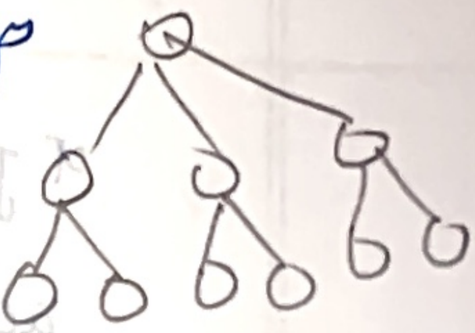
* It consists of
  Nodes connected
  nodes connected b
  nodes.
* There is a structu
* Further divided into

* It uses Focus + C
  Consideration
* Both local and global

* The nodes then position themselves in various iterations
* This process is continued until the forces attracting the nodes remain unchanged
* It makes better use of the space layout
* It take $O(N \log N)$ or $O(E)$ time and is therefore not suitable for more than 100 nodes.

Tree Layout:

* A tree layout is more pranjs to human's eye than a normal graph layout



* It consists of a root node, its ~~Childnode and the~~ nodes connected to it as children nodes and the nodes connected to the children node as grandchildren nodes.
* There is a structure and a context in this layout
* Further divided into
  - Hyperbolic tree
  - Radial tree.
  - H3 visualization
* It uses "focus + context + animation" idea in h consideration
* Both local and global view in 2D and 3D is possible

8. User centric ~~functionalities~~ visualization:

* Here the characteristics of a person is presented and also the subjects and relationships of interest are presented

* The user centric ~~functionality~~ visualization could include

⇒ groups / users with similar features
⇒ key actors with high impact
⇒ people with high social relationships or active social interactions.

* The user centric approach allows people to access and discover social networks of their choice.

9.

Content Centric Visualization:

* Various different types of contents are provided to the people who analyze social network

↳ distribution of users ⇒ key user opinions and the controversial comments

↳ Users who big high impact to social communities, especially in the near future

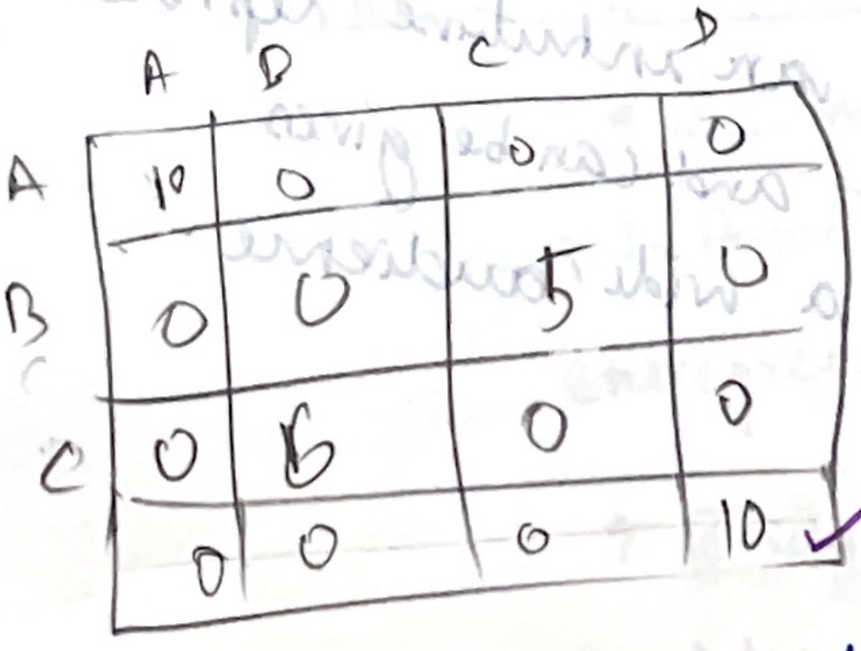
↳ Relationships between different content groups.

Hybrid Visualization :

* Usually social network is really diverse and cannot be done with only few approaches
* There is a need for hybrid approach that combines different aspects of attributes like Content and people
* This is especially of importance in daily and Email services.

Matrix based

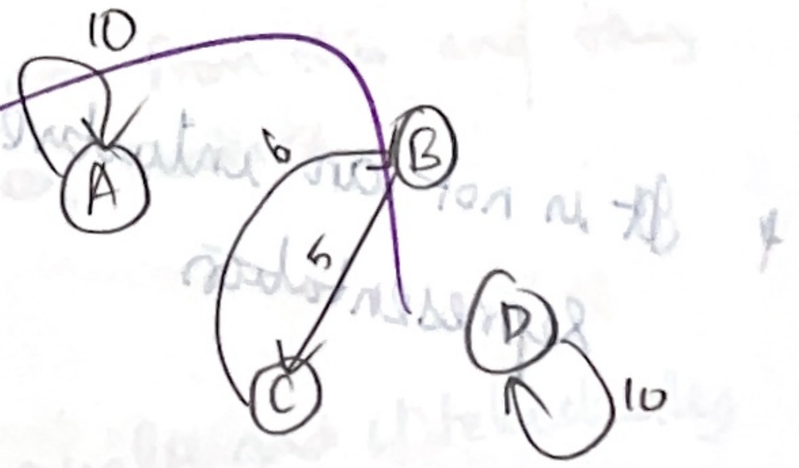
* Nodes are represented in rows and columns
* The values between the cell represent connection between the nodes.



The weights are specified in the matrix

Node link Digram

* Nodes are represented as labelled circles
* Edges are lines joining the nodes and the weights are present on the links



* Matrix representation is suitable for large networks

* Used for dense networks

* Has low initiation time for explanation

* cannot be used for path finding

* No problem of ~~edge~~ node overlapping and cross links

* It is not an intuitive representation

* Node link representation is suitable for small networks or for compact representation of networks

* Generally suitable for sparse networks

* Has high initiation time for exploration

* can be used for path finding

* Node overlapping and cross links make it difficult to understand the representation (if) it becomes unreadable

* It is an intuitive representation ~~that~~ and can be given for a wide audience.

PART-A:

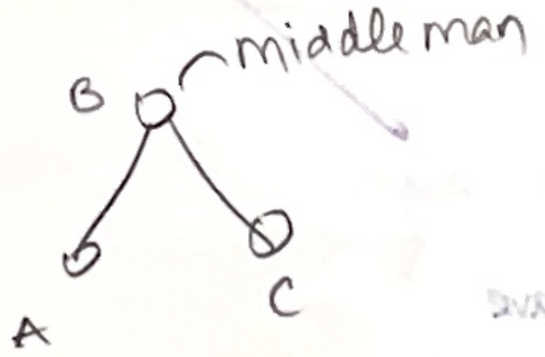
1. A triad is a set of 3 nodes that could be interlinked in any ways. If directed there are 16 different ways of arranging 3 nodes.

Triadic analysis is to find how many times a node has repeated one of the 16 patterns in its network.

eg The hijacked pilot of 9/11 attack had 300 such connections.

---

2. A forbidden triad is the one in which 2 ends don't communicate directly with each other and requires a middleman.



If A and C has to communicate, it has to go through B.

Significance of forbidden triads in network

* This middleman could gain from this and they can take forms as broker for houses, bankers, entrepreneurs etc.

* Bridging gap between socially and intellectually different communities

eg. Musicians, scientists, musicians and scientists.

3. Visualization is an important concept in SNA.

* It helps in linking human vision to computer
* It helps in pattern finding
* It helps to get meaningful insights out of data
* It helps in finding structural information.

---

4. For matrix based visualization, the following are best suited :-

* Large Networks that require low imitation time.
* Networks with node overlapping
* Networks with cross links
* ~~Un~~ Directional data
* ~~Hig~~ High Density networks
* Networks that do not have path finding application

---

5. The 3 popular centrality measures are:

Degree, Betweeness, closeness.

Degree Centrality * It is the number of edges a node is incident on.

* For directed graph in-degree and out-degree centrality is present

~~* For undirected, it is degree Centrality~~

Betweenness Centrality: * It is a measure of the extent to which a node is present between other nodes

* The extent to which it acts as a bridge node for acts as the shortest path.

* High betweenness ⇒ crucial for information passing

Closeness Centrality:

* It is a measure of how near other nodes are to the node.

* Calculated as mean shortest path

* If the closeness centrality is high, then it is used for distribution of information and is a very crucial node.

Advantages of Node Link Diagrams:

* It can be presented to a wider audience as the representation is intuitive.

* Can be used for abs. small / sparse networks

* Can be used for Compact representation of networks

* Can be used for path finding applications.