COMPUTER ENGINEERING DEPARTMENT

BDA Assignment 2

COURSE: **B.E.** YEAR: **2020-2021** SEMESTER: **VII**
DEPT: **Computer Engineering**
SUBJECT CODE: **CSDLO7032** DATE OF ASSIGNMENT: **08-10-2021**
==========

NAME: **AMEY MAHENDRA THAKUR** ROLL NO.: **50**

CLASS: **COMPS BE B** DATE OF SUBMISSION: **08-10-2021**

Sr. No.	Questions
1	What is a Data Stream Management System? Explain with Block Diagram.
2	Why is finding similar items important in Big Data? Illustrate using two example applications.
3	Explain the Girvan-Newman algorithm to mine Social Graphs.

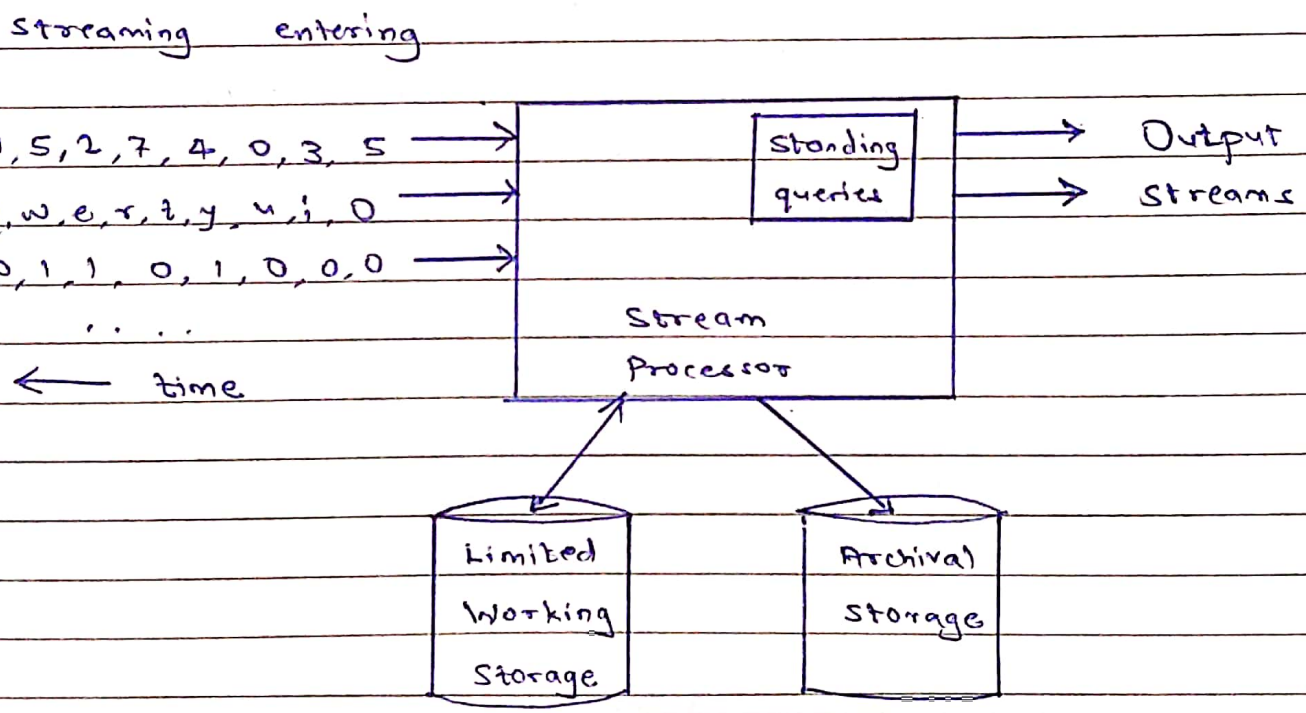
Signature of Student

AMEY THAKUR B-50 Amey

Q1. What is Data Stream Management system? Explain with Block Diagram.

Ans:

- A DSMS is a computer software system to manage continuous data streams. It is similar to DBMS, which is, however, designed for static data in conventional databases.
- A DSMS also offers flexible query processing so that the information needed can be expressed using queries. It executes a continuous query that is permanently installed.
- Since most DSMS are data-driven, a continuous query produces new results as long as new data arrive at the system.



AMEY THAKUR B-50 Amey

- Any number of streams can enter the system.
- Each stream can provide elements at its own schedule; they need not have the same data rates or data types and the time between elements of one stream need not be uniform.
- The fact that the rate of arrival of stream elements is not under the control of the system distinguishes stream processing from the processing of the data that goes on within a database management system.
- The latter system controls the rate at which data is read from the disk and therefore never has to worry about data getting lost as it attempts to execute queries.
- Streams may be archieved in a large archival store, but we assume it is not possible to answer queries from the archival store.
- It could be examined only under special circumstances using time-consuming retrieval processes.
- There is also a working store, into which summaries or parts of streams may be placed and which can be used for answering queries.
- The working store might be disk, or it might be the main memory, depending on how fast we need to process queries.
- But either way, it is of sufficiently limited capacity that is cannot store all the data from all the streams.

AMEY THAKUR B-50 Amey

Q2. Why is finding similar items important in Big Data? Illustrate using two example applications.

Ans:

① Plagiarism:

- Finding plagiarised documents tests our ability to find textual similarity.
- The plagiarizer may extract only some parts of a document for his own.
- He may alter a few words and may alter the order in which sentences of the original appear.
- Yet the resulting document may still contain 50% or more of the original.
- No simple process of comparing documents character by character will detect a sophisticated plagiarism.

② Articles from the same sources:

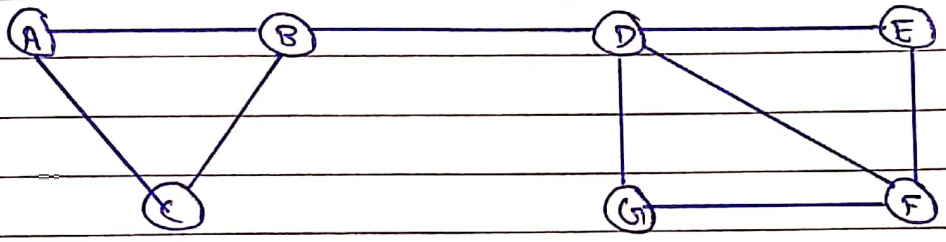
- It is common for one reporter to write on a news article that gets distributed, say through the associated press, to many newspapers, which then publish the article on their website.
- Each newspaper changes the article somewhat.
- They may cut out paragraphs, or even add material of their own.
- They most likely will surround the article with their own logo, ads and links to other articles at their site. However, the core of each newspaper's page will be the original article.
- News aggregators try to find all versions of such an article, in order to show only one and that task requires finding when two web pages are textually similar, not identical.

AMEY THAKUR B-50 Amey

Q3. Explain the Girvan-Newman algorithm to mine social graphs.

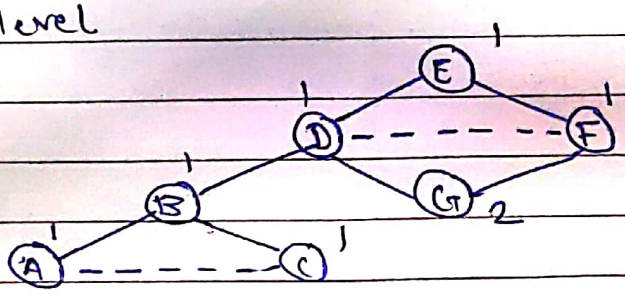
Ans:

- In order to find out between edges, we need to calculate shortest paths from going through each of the edges.
- Girvan-Newman algorithm visits each node X once and computes the number of shortest paths from X to each of the other nodes that go through each of the edge.
- The algorithm begins by performing a Breadth First Search (BFS) of the graph, starting at node X.
- The edges that go between node at the same level can never be a part of a shortest path from X.
- Edges DAG edge will be part of at least one shortest path from root X.
- Consider the following graph:



Sample Graph

Following figure shows the BFS starting at node E. Solid edges are DAG and dashed edge-connect nodes on the same level



(Step 1 & 2 of GN algorithm)

AMEY THAKUR B-50 Amey

- The second step of GN algo is to label each node by the number of shortest paths that reach it from the root.
- Start by labeling the root 1, then, from the top down label each node Y by the sum of the labels of its parents
- The labeling of nodes is shown in above figure.
- The final step is to calculate for each edge e the sum over nodes Y of the fraction of shown paths from the root x to Y then go through e.
- Each node other than the root is given a credit 1.
- This credit may be divided among nodes and edges above.

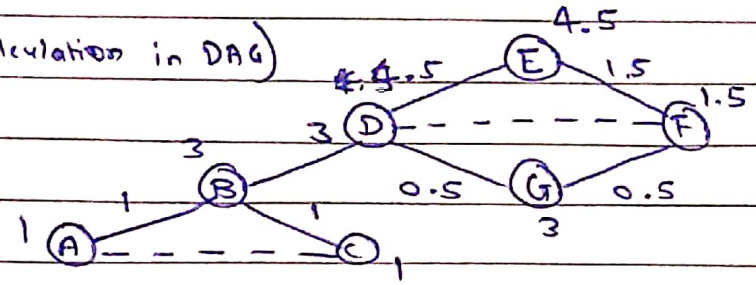
The rules for calculation are as follows:

- Each node in DAG gets a credit.
- Each non-leaf node gets a credit equal to 1 plus the sum of the credits of the DAG edges from that node to the level
- A DAG edge e entering node Z from the level above is given a share of the credit of Z proportionate to the fraction of shortest paths from the root to Z that go through e.
- After performing the credit calculate with each node as the root, we sum the credits for each edge.
- As each shortest path will have been discovered twice, we must divide the result / credit for each edge by 2.

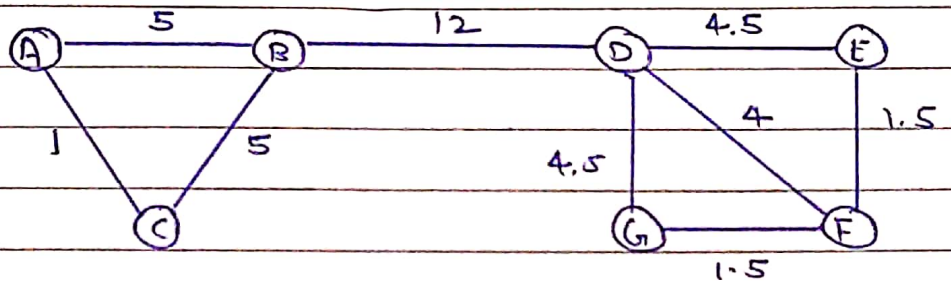
AMEY THAKUR B-50 Amey

- Following graph shows the calculation for DAG starting from node E.

(Calculation in DAG)



- To complete the betweenness calculation, we have to repeat this calculation for every node as the root and sum the contributions.
- After calculations, following graph shows final betweenness values.



(Graph with betweenness value)

- We can cluster by taking the in order to increasing betweenness and add them to the graph at a time.
- We can remove edge with 'highest value to cluster the graph
- In the above example, we remove edge BD to get two communities as follows'.

Communities by clustering