Computing Lab II Spring 2020
Final Assignment

Instructions:

1. It consists of three sections namely C++,SQL and NoSql.
2. Each section is having 2 questions each.
3. You need to do **any one of them from each section**,in total 3 questions each one coming from 3 different sections.
4. Plagiarism Rule: If your code matches (**more than 50%**) with another student's code, all those students need to provide clarification for that,if the justification is not found convincing enough they will be awarded with **Zero marks**. Therefore, it is your responsibility to ensure that neither you copy anyone's code nor anyone is able to copy yours.
5. Code error: If your code doesn't run or produces error while running, you will be awarded **Zero marks**.

Deliverables:

1. All python codes (mapper,combiner and reducer),result.txt,readme+makefile for Nosql. Put them in folder named **<Roll_no>_nosql**.
2. C++ code with makefile for the C++ section.Put them in folder named **<Roll_no>_cpp**.
3. Separate text file containing sql query and output.Put them in folder named **<Roll_no>_sql**.
4. Zip all 3 folders in zipfile named **<Roll_no>_final_assignment.tar.gz** and submit.

Deadline: **7th April,2020 (Tuesday)**.

C++ Section:

Question 1.

Given a directed graph, write a C++ program to check whether the graph contains a cycle or not. Your function should return true if the given graph contains at least one cycle, else return false. For example, the following graph contains three cycles 0->2->0, 0->1->2->0 and 3->3, so your function must return true.

Input: 1. number of nodes N, and number of edges E where nodes are numbered 0, 1, 2, … N-1
2. edge list in the form of E ordered pairs "a b", (a,b <N) one per line where a and b are the node IDs.

Output: If a cycle exists, it should print "Graph contains cycle", otherwise it should print "Graph doesn't contain cycle"

Examples:

Input:

4 6
0 1
0 2
1 2
2 0
2 3
3 3

Output: Graph contains cycle

Input:

3 3
0 1
0 2
1 2

Output: Graph doesn't contain cycle

Question 2:

Given a connected, directed graph. Each node is connected to exactly two other nodes. There is a weight associated with each edge denoting the cost to reverse its direction. Write a C++ program to find the minimum cost to reverse some edges of the graph such that it is possible to go from each node to every other node (i.e. to create a directed cycle).

Input: 1. number of nodes N and number of edges E where nodes are numbered 0, 1, 2, … N-1

2. weighted edge list in the form of E ordered tuples "a b w", (a,b <N) one per line where a and b are the node IDs and w is the weight (it is assumed that each node will have exactly two edges, which may be either incoming or outgoing or both so that each node ID appears exactly twice in the edge list).

Output: The minimum cost value

Examples:

Input:
5 5
1 2 7
5 1 8
5 4 5
3 4 1
3 2 10

Output: 15

Input:
6 6
1 5 4
5 3 8
2 4 15
1 6 16
2 3 23
4 6 42

Output: 39

SQL Section:

Question 1:

Schema Details:

Customer

CustomerId	FirstName	LastName	City	Country	Phone

Orders

OrderId	OrderDate	OrderNumber	**CustomerId**	TotalAmount

OrderItems

ItemId	**OrderId**	**ProductId**	UnitPrice	Quantity

Supplier

SupplierId	CompanyName	ContactName	ContactTitle	City	Country	Phone	Fax

Product

ProductId	ProductName	**SupplierId**	UnitPrice	Package	IsDiscontinued

Problem Statement:

Find the most productive month for supplier company named “Leka Trading” in year 2013.
List CompanyName, Month, TotalAmount earned that month.
Month should be in format **MON.** Eg (JAN for January, FEB for February...)

Question 2:

Schema Details:

Customer

CustomerId	FirstName	LastName	City	Country	Phone

Orders

OrderId	OrderDate	OrderNumber	**CustomerId**	TotalAmount

OrderItems

ItemId	**OrderId**	**ProductId**	UnitPrice	Quantity

Supplier

SupplierId	CompanyName	ContactName	ContactTitle	City	Country	Phone	Fax

Product

ProductId	ProductName	**SupplierId**	UnitPrice	Package	IsDiscontinued

Problem Statement:

For each Customer Id, find the month name in which he has purchased maximum no of items along with no of items purchased. Output should be in descending order of No of Items purchased.
List CustomerId, Month (format 'January', 'February'...), Total Items Purchased.

NoSql Section:

Question1:

The matrix operation task

You have been given the matrices A,B,C,D,E in the dataset. Each of the matrices are of dimension (50 × 50). You have to write a mapper, a combiner and a reducer to get the following.

$$(D-B-C+A) \times E$$

Question 2:

The matrix operation task

You have been given the matrices A,B,C,D,E in the dataset. Each of the matrices are of dimension (50 × 50). You have to write a mapper, a combiner and a reducer to get the following.

$$E \times (B-A-D+C)$$

Important points to keep in mind:

1. You can combine the outputs of multiple commands using ';' and '(' ')' like below.
   (python mapper.py file1.txt | sort | combiner.py; python mapper.py file2.txt | sort | combiner.py; python mapper.py file3.txt | sort | combiner.py;) | sort | reducer.py > result.txt
2. The mapper code should take only the one file name as argument (as explained in the last point) and should go through the corresponding file content only once. You can assume the files to be available in the same directory where your codes are present, i.e., "./".
3. Your combiner can't just be receiving and sending the same stream, it must perform some reduction. Neither the combiner nor the reducer should read any of the files. You can't save any intermediate result as you are writing only one set of mapper, combiner, reducer.
4. Saving the final results: The final result should be saved as "result.txt". The format should be like below.
   <row_number>,<column_number>,<value>

Dataset

You can download the dataset in the below link.

https://drive.google.com/drive/folders/1ajfvkDRVU8lvU9Fu8_YB7es_S3bj_eSP?usp=sharing

The dataset contains 3 files named “file1.txt”, “file2.txt”, “file3.txt”. Each of the files have the data in the following format.

<matrix_name>,<row_no>,<column_no>,<value>

e.g., A,2,3,-5.23

(this entry means that the element in the 2nd row and 3rd column of matrix A has the value -5.23)

- The data is available in a dense format. i.e.,there will be entries for all the elements with value 0 too.
- You may not have all the elements of any particular matrix in one single file. They might be distributed in all the three files.
- The entries in the files may not be in sorted format.