DATA SYSTEMS

Assignment 4

Deadline: 16th March 11:55 P.M.

Instructions:

1. You are not allowed to use any external library/jar files in this assignment.
2. Plagiarism will not be tolerated. **Copying from any source and in any form (friends/seniors/internet) will fetch you straight 0 marks in all the assignments and quiz.**
3. Be careful about your submissions. You must strictly follow the upload format. Failure to do so will fetch you a zero.
4. Languages allowed to code are C/C++/Java/Python

Given M memory blocks and two large relations R(X,Y) and S(Y,Z). Develop iterator for the following operations.

- **SortMerge Join**
  - **open()** - Create sorted sublists for R and S, each of size M blocks.
  - **getnext()** - Use 1 block for each sublist and get minimum of R & S. Join this minimum Y value with the other table and return. Check for $B(R)+B(S)<M^2$
  - **close()** - close all files
- **Hash Join**
  - **open()** - Create M1 hashed sublists for R and S
  - **getnext()** - For each Ri and Si thus created, load the smaller of the two in the main memory and create a search structure over it. You can use M1 blocks to achieve this. Then recursively load the other file in the remaining blocks and for each record of this file, search corresponding records (with same join attribute value) from the other file.
  - **close()** - close all files

Join condition (R.Y==S.Y).

Use 1 block for output which is filled by row returned by getnext() and when it gets full, append it to the output file and continue.

Input Parameters:

1. Path to file containing relation R

2. Path to file containing relation S
3. Type of join sort/hash
4. Number of blocks M

Attribute Type :

Note that all attributes, X, Y and Z are strings and Y may be a non-key attribute.

Block Size :

Assume that each block can store 100 tuples for both relations, R and S.

Input format :

Create a bash script named <RollNumber.sh> to compile and run your code. The command for execution would be of the form:

<RollNumber.sh> <path of R file> <path of S file> <sort/hash> <M>

Output file: <R filename>_<S filename>_join.txt (Kindly note it should contain only R & S filename and not their path).

Graph: Vary M from 50 to 100 blocks in steps of 10 blocks and calculate the execution time. Plot the graph of M versus (X-axis) execution time (Y-axis) separately for sort merge join and hash join.

Submission :

Create a folder with the name rollno_Assign4 and put the following into it -

1. Your source code in the folder named as code . (It should not contain the code used for graph generation).
2. A pdf file with the name analysis.pdf containing the following information.
    a. Configuration of the System
    b. Both matrix(containing M, time data) used to plot the graph (in tabular format)
    c. Both the image files of the graph
3. A bash file with the name RollNumber.sh that compiles and runs your code. Compress the folder and upload the rollno_Assign4.tar.gz