DATABASE SYSTEMS
Assignment 4
Deadline: : 9:00PM, 10th November, 2018

Instruction:
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 quizzes.
3. Be careful about your submissions. You must strictly follow the upload format. Failure to do so will lead you to lose marks.
4. Languages allowed to code are C/C++/Java.

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

- Sort Merge join
    - open() Create sorted sublists for R and S, each of size M buffers.
    - getnext() Use 1 block for each sublist and get min. of R & S. Join this minimum Y value with other table and return. Check for $B(R)+B(S)<M^2$
    - close() Close all files
- Hash Join
    - open() Create M hashed sublists for R and S
    - getnext() For each R_i and S_i thus created, load the smaller of the two in the main memory and create a search structure over it. You can use M buffers to achieve this. Then iteratively load the other file in the remaining buffer 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 buffer for output which is filled by row returned by getnext() and when it gets full, append it to output file and continue.

Input Parameters:
You will be given as an input the files containing relations R and S and the value of M blocks.

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:
 ./a.out <path of R file> <path of S file> <sort/hash> <M>

Output:
Output File:
<R filename>_<S filename>_join (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.
You have to plot the graph of M (X axis) versus 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 following information
   a) Configuration of the System
   b) Both matrix used to plot the graph (in tabular format)
   c) Both the image files of the graph
3) A bash file with the name rollno.sh that take to compile your code and run your code. Compress the folder and upload the rollno_Assign4.tar.gz