CS 4700/6700 Project P1

In this project each student will implement a Java program or a Python program to process relational algebra (RA) queries. Combined with file editing (for updating data), this can lead to a simplified database system.

Input relations/tables are given files in csv format; their file names have the form RN.txt, where RN is a relation's name. The first line of each such file lists the attributes of the relation, separated by commas. Attributes are assumed to have either strings or integers or decimal numbers as their domains. (In Java, students can check what domain a given attribute has from the attribute values in the input files.)

Relational algebra queries are given in the input file (namely RAqueries.txt), having one query per line. Your program reads RAqueries.txt and then processes the queries contained in it. Your program needs to write to an output file named as "RAoutput.csv"; it contains one RA query on one line, followed by the output produced for that query. (Add two blank lines before starting output for the next query.) The file should contain all the queries your program can process. Your program will also write a shortened version of what was written to the output file, listing the first 3 lines of output for each query. Your program will determine the relation names of interest by reading RAqueries.txt.

Inside RAqueries.txt, relational algebra operators PROJECT, SELECT, INTERSECT, and JOIN are respectively typed as these four-letter strings: PROJ, SELE, INTE, JOIN; Natural JOIN is typed as *, UNION is typed as U, CROSS PRODUCT is typed as X, and DIFFERENCE is typed as -; subscripts are used to represent conditions for selections and to represent list of attributes for projections; the comma "," is used to represent "AND" and the word "OR" represents itself. For simplicity, in this project we do not work with the logical "OR" and "NOT" and, among the joins, we only work with the natural join.

csv stands for "comma separated values". An example csv line with 5 values is:

2, abc, 4, xyzu, 9.3

An example RAqueries.txt contains the following four lines/queries:

SELE_{Payment > 70} (Play)

PROJ_{ANO} (ACTORS * Play)

(PROJ_{ANO} (SELE_{Payment > 70} (Play))) - (PROJ_{ANO} (SELE_{Payment < 60} (Play)))

(PROJ_{ANO} (SELE_{Payment > 90} (Play))) U (PROJ_{ANO} (SELE_{ANAME='Swanson'} (ACTORS)))

Students can use any structures/classes of Java or Python as long as they are not intended for directly supporting RA operators. Students cannot use database-like systems/classes in their programs. You should implement functions for various RA operators and then use the functions to process the queries. Your program can assume that all the input files are in the folder where your submitted program is to be run.

Students write a report to describe the following:

1. A line (or two) that can be copy/pasted to Run the Program on Windows or Unix; type **How to Run the Program** as section title of this line.
2. A list of **Implemented Operators/Queries** (using Line 1, … Line 4 to refer to the four queries and type PROJECT, SELECT, INTERSECT, JOIN, *, UNION, Difference, CROSS PRODUCT, Composition to indicate the list of operators were implemented). Composition (of operators) is considered as an operator.

3. **One Screenshot** showing results produced by the implemented program when processing line x in RAqueries.txt, where x is the last line your program can process. Demonstration on how to make screenshots is given in a file in the project folder on pilot.
4. **Output Produced** by the implemented program answering the RA queries in RAqueries.txt (these are the contents of RAoutput.csv).

Use the bolden words in each item above as a section title in the report to facilitate reading/marking. Use the string "Section: " before the section titles. Sections not included in your report in this manner will be treated as "not submitted". The four sections should be in the order given above, and the first two sections need to appear on the first page.

Submit a zipped folder on pilot containing: your report (a file) and your program (a compressed file using zip etc). Your submitted files should not contain the example files provided by the instructor. Your program file folder can contain subfolder but not sub-subfolders. Use P1ReportLN.X as the name of your report, where LN is your last name, and X can be pdf, doc, or txt.

Marking will focus on what was done, and correctness and readability of the submitted program.

Different input files may be used when marking your program.

Cheaters will be penalized and will be reported to the university for disciplinary action. Fake program output (e.g. writing the output without implementing the relational algebra operations) is considered a form of cheating.