CSE6332 Cloud Computing & Big Data Fall 2022

Midterm Exam

October 11 2022

Answer all questions in the space provided next to the question. Keep your answers brief and precise. It is open notes but closed books. You are allowed to bring up to 40 pages of notes stapled together in **one** folder. You are not allowed to remove any of the pages from your notes or the exam paper during the exam. You are allowed to use pens, pencils, and erasers only. You may use the back pages of the exam for scratch paper. 100 points possible. This exam is worth 20% of your final grade. Duration: 1 hour and 20 minutes. Good luck!

FIRST-NAME LAST-NAME:

STUDENT ID:

1. *(25 points)* Suppose that we have a dataset that contains user ratings of movies (from 1 to 5). This dataset is stored in an HDFS text file with lines of the following format:

   ```
   MovieID,UserID,Rating,Date
   ```

   where `MovieID` is the ID of a movie (a long int), `UserID` is the ID of a user (a long int), `Rating` is the user rating of this movie (an int from 1 to 5) on the given `Date`. Write Map-Reduce pseudo-code to do the following: For each different movie, return the total number of ratings for this movie and the average rating of this movie (a double between 1.0 to 5.0). The output should be:

   ```
   MovieID Count AvgRating
   ```

   (separated by tabs or spaces) where `Count` is the total number of ratings and `AvgRating` is the average rating of the movie `MovieID`. You may use the Java class `Pair` used in Project 2. You need to write your mappers and reducers of your Map-Reduce jobs in pseudo-code (not Java). You don't need to write the main program or the run method.

2. *(40 points)* In Project 2, we defined a block matrix to be a dataset of blocks. Each block is a dense Java square matrix of size rows×columns constructed with the Java class Block, where rows=columns=100. A block matrix is stored in HDFS as a binary file (in SequenceTextInputFormat) of key-values, where the key is a pair of block coordinates (c_i, c_j) (constructed with the Java class Pair) and the value is a block. A matrix element M_{ij} is stored inside the block with block coordinates (i/rows,j/columns) at the location (i%rows,j%columns) inside the block.

   (a) *(25 points)* Write Map-Reduce pseudo-code to convert a block matrix to a sparse matrix. Here, a sparse matrix is an HDFS binary file (in SequenceTextInputFormat), where the key is Pair(i, j) and the value is v, where v is that matrix value at the indices i and j.

   (b) *(15 points)* Write Map-Reduce pseudo-code that returns the transpose of a block matrx. The transpose must also be a block matrix. The transpose of a matrix M is M^T such that $M^T_{ij} = M_{ji}$.

   You need to write your mappers and reducers of your Map-Reduce jobs in pseudo-code (not Java). You don't need to write the main program or the run method.

3. *(35 points total)* Suppose that we represent a flow network using a directed graph. A directed graph is represented as a set of edges, where an edge `(i,j,f)` from the vertex `i` to the vertex `j` represents a flow of size `f`. An edge is stored in the input text file `graph.txt` as a text line `i,j,f`. You may assume that `i` and `j` are integers and `f` is a positive double. The flow can be visualized as a physical flow of a fluid through the network, following the direction of each edge. The conservation constraint for flows says that the amount that flows into a vertex equals the amount flowing out of this vertex. Write Map-Reduce pseudo-code that returns all nodes that violate the conservation constraint, that is, the nodes whose total incoming flow is different from the total outgoing flow. The output should contain the id of each such node and its total flow difference (the outgoing flow minus the incoming flow). You only need to write the mappers and reducers of your Map-Reduce jobs using pseudo-code (not Java). You don't need to write the main program or the run method.