Programming Project 1 for Distributed Systems (ECE 499/599 / CS 419/519)

1 Overview

The learning objective of this project is for students to get familiar with using Map-Reduce paradigm for solving problems. After finishing the assignment, in addition to improving their understanding of Map-Reduce programming paradigm, you should be able to identify what problems lend themselves well to this paradigm and write programs using Java/Apache Hadoop and solve problems using this paradigm.

2. Submission Guidelines

You need to submit code to solve the problem(s) described below and then demo your code to the TA.

3 Lab Environment and Tasks

Installing Apache Hadoop: In this MP, you will use Apache Hadoop and Java. Please install the latest version of Apache Hadoop compatible with your platform and necessary libraries and files to program using Java and Hadoop.

Here is a URL to help you get started but you are welcome to use any other resource:
https://hadoop.apache.org/docs/r1.2.1/single_node_setup.html

While the instructions above do include Windows environments, it is highly recommended that you do the assignment in a Linux environment.

3.1 [40 pts] **Map-Reduce Task 1:** Write a Map-Reduce program to dictionary sort the words in a given text file and out put a sorted list of words. Each word is preceeded by a serial number. When words appear multiple times list them once but include the number of times they appeared (see example below). At the end of the output file list the number of unique words and total number of words.

Input File: Will be a .txt file containing english text. The length need not be fixed.
Example: `The cow jumps over the moon.`

Output File: Dictionary sorted list of words as shown below
Example:

```
1 cow
2 jumps
3 moon
4 over
5 the 2
Unique Words: 5
Total Words: 6
```

Grading:
Program compiles and runs without failure: 10 points
Passes two test cases: 15 points each

3.2 [60 pts] **Map-Reduce Task 2:** BeaverMart a big supermarket chain wants to find out what items in its stores are bought together so it can optimize the store layouts. It has records of customer purchases tracked using store reward cards. Each record is a tuple of items that are bought in a single transaction (e.g., {item1, item2, item3, …}). Write a Map-Reduce program to compute how many times a pair of items are bought together.

Input File: Will be a .txt file containing records, one per line. The length of the file need not be fixed.
Example:

```
Whitey Toothpaste, Best Bread, Fluffy Pizza, BeavMoo Milk
Apples, BeavMoo Milk, Bananas, Best Bread
```

Output File: Pairs of items along with the number of times they have been purchased together
Example:

```
(Whitey Toothpaste, Best Bread) 1
(Whitey Toothpaste, Fluffy Pizza) 1
(Whitey Toothpaste, BeavMoo Milk) 1
(Whitey Toothpaste, Apples) 0
(Whitey Toothpaste, Bananas) 0
(Best Bread, Fluffy Pizza) 1
(Best Bread, BeavMoo Milk) 2
(Best Bread, Apples) 1
(Best Bread, Bananas) 1
(Fluffy Pizza, BeavMoo Milk) 1
(Fluffy Pizza, Apples) 0
(Fluffy Pizza, Bananas) 0
(BeavMoo Milk, Apples) 1
(BeavMoo Milk, Bananas) 1
(Apples, Bananas) 1
Total Pairs: 15
```

Grading:
Program compiles and runs without failure: 10 points
Passes two test cases: 20 points each
Design (how fast and scalable is your design): 10 points