# Big Data Analytics

**Homework 4 (MapReduce)**

Name: - Sreenath Reddy kurukunda  
ID:- 811211580

In this homework, there are 4 questions + 1 bonus question, covering the topic of MapReduce. If you can answer the bonus question correctly, you can obtain 20 extra points. The maximum mark for this homework is **120 points**, which will be later scaled.

1. Please use your own words to describe the Map and Reduce functions. [20 points]

**ANSWER:**

The Map and Reduce functions can handle huge data by distributing them throughout various nodes and then combining or decreasing the findings of a nodes. It's a programming framework. It includes three steps: Reduce, shuffle, and map.  
The key value pair is used as the input and the output types for the map function. It is useful for finding information in large databases. It employs an iterative strategy for searching the data in the records.

The map function collects data from the user and stores as a key-value pair. Reduce function gets the data from mapper. To process the data, the logic was written in the Map function. It combines all the orders and gives the user the result.

2. Assume that we have a relational database *D* containing tuples *t* with attributes *t*[*A*]. Write the pseudo code of Map and Reduce functions for retrieving all tuples with attribute *t*[*A*] > 100. [20 points]  
**ANSWER:**

The pseudo code for the relational database D with the tuples t and with attribute t[A].

Map(t)

{

if(t[A] > 100)

{

intermediate(t, 1)

}

}

Reduce (t, count)

{

output(t)

}  
  
Map function.  
The Map function takes a tuple t from the database D and checks if the value of attribute t[A] is greater than 100. If it is, the function emits a key-value pair (t, 1) where the key is the tuple itself and the value is just a dummy value of 1.   
  
Reduce function:  
The Reduce function takes the key-value pairs emitted by the Map function as input. Since we only emitted the tuples that satisfy the condition in the Map function, this will give us the desired output - all tuples with attribute t[A] > 100.

3. Assume that we have two relational databases *R* and *S* containing tuples *r* and *s*, respectively. Write the pseudo code of Map and Reduce functions for joining databases *R* and *S* on attribute *A* (i.e., joining tuples *r* and *s* satisfying *r*[*A*] = *s*[*A*]). [20 points]

**ANSWER:**The pseudo code for joining the tuples r and s to satisfy r[A] = s[A] can be written as,  
  
Map (inKey, inValue)

{

If ( inKey == D[A] )

{

intermediate (key, value)

}

}

Reduce (key, value)

{

For each value in D

{

If (R[A] == S[A])

Return (value)

}

}

4. Please write a tutorial for installing Apache Hadoop (on either windows or MacOS) or configuring Amazon Web Services. *Please use screen captures and record the entire installation process step by step.* [40 points]  
  
**ANSWER:**To install Apache Hadoop on macOS.  
  
1) Go to Apache Hadoop website, and download the latest Hadoop stable version.  
Graphical user interface, text, application, email

Description automatically generated

2) Click on the extension [hadoop-3.3.4.tar.gz](https://dlcdn.apache.org/hadoop/common/hadoop-3.3.4/hadoop-3.3.4.tar.gz), and download the file.  
Graphical user interface, application

Description automatically generated

3) Extract the files from the downloaded file.

A picture containing text, electronics, screenshot

Description automatically generated

**4)** Set the java and Hadoop path in the environment variables.Text

Description automatically generated  
  
5) When the path is set you can see the Hadoop version without any error.

**Text

Description automatically generated**6)Now we need toaddthe property in hdfs-site.xml file.  
  
**Text

Description automatically generated**

7) Now we need to addthe property in core-site.xml file.**Text

Description automatically generated**  
  
8) Now we need to addthe property in mapred-site.xml file.

**Text

Description automatically generated  
  
9)** Now we need to addthe property in yarn-site.xml file **Text

Description automatically generated**

**Bonus Question [20 extra points]**

5. Implement the Word Count example (as mentioned in the lecture slides) on Hadoop or AWS. *Please include the source code and add screen captures to illustrate the execution process.*

**ANSWER:**

package wc;

import java.io.IOException;

import java.util.\*;

import org.apache.hadoop.\*;

public class WordCount extends Configured implements Tool {

public static void main(String args[]) throws Exception {

int runResult = ToolRunner.run(new WordCount(), args);

System.exit(runResult);

}

public int run(String[] args) throws Exception {

Path inputPath = new Path(args[0]);

Path outputPath = new Path(args[1]);

Configuration config = getConf();

Job countFile = new Job(config, this.getClass().toString());

FileInputFormat.setInputPaths(countFile, inputPath);

FileOutputFormat.setOutputPath(countFile, outputPath);

countFile.setJobName("WordCount");

countFile.setJarByClass(WordCount.class);

countFile.setInputFormatClass(TextInputFormat.class);

countFile.setOutputFormatClass(TextOutputFormat.class);

countFile.setMapOutputKeyClass(Text.class);

countFile.setMapOutputValueClass(IntWritable.class);

countFile.setOutputKeyClass(Text.class);

countFile.setOutputValueClass(IntWritable.class);

countFile.setMapperClass(Map.class);

countFile.setCombinerClass(Reduce.class);

countFile.setReducerClass(Reduce.class);

return countFile.waitForCompletion(true) ? 0 : 1;

}

public static class Map extends Mapper<writeLong, text, text, writeInt> {

private final static writeInt one = new writeInt(1);

private text word = new text();

@Override

public void map(writeLong key, textValue, Mapper.Context context) throws IOException, InterruptedException {

String readLine = value.toString();

StringTokenizer tokenizer = new StringTokenizer(readLine);

while (tokenizer.hasMoreTokens()) {

word.set(tokenizer.nextToken());

context.write(word, one);

}

}

}

public static class Reduce extends Reducer<text, writeInt, text, writeInt> {

@Override

public void reduce(textKey, Iterable<writeInt> values, Context context) throws IOException, InterruptedException {

int sum = 0;

for (writeInt value : values) {

sum += value.get();

}

context.write(key, new writeInt(sum));

}

}

}

## Submitting Your Assignment

*All work must be your own. Copying other people’s work or from the Internet is a form of plagiarism and will be prosecuted as such.*

You may submit a Microsoft Word (.docx) document as an attachment. If you attach a document for your assignment, be sure to include your name in the text of the document and in the name of the document.

You can submit multiple times and only the last submission attempt will be considered for grading.

* Submissions sent by email will NOT be accepted.