Following experiments are designed to be done on Ubuntu 12.04 LTS Linux installed Virtual machines.

1. Video conversion performance comparison (30%)

The task to be performed on the virtual machines is as follows: to convert a given video file to reduce the size and quality. You need to run the above conversions on different sizes of VMs and record the conversion time. You need to run the experiment on at least three different sizes of VMs, and at least four different values per parameter (quality value, Frame Rate.. etc) when convert using the Handbreak tool. You may choose at least one variable video parameter when converting, to show the change in size and quality of the video better.

Learn more about the quality parameter in HandBreak here; https://trac.handbrake.fr/wiki/ConstantQuality

You need to prepare a report to cover the following content:

a) to record the conversion time for each conversion on each flavor (10%)

b) using the collected data to describe your observation and to explain why such observations (15%)

c) extra 5% (out of 30%) marks will be given to additional presentations of the results other than tables to better show the comparisons

Guidance:

Download Video file on to the instance.

Install Handbreak video transcoder to convert the video files

```
$ sudo add-apt-repository ppa:stebbins/handbrake-releases
$ sudo apt-get update sudo apt-get install handbrake-cli
```

Example: Using the following command you can convert video files by specifying the video quality.

```
$ HandBrakeCLI -i $input_file_name -o $outputfile_name –q
quality_value
```

You may experiment with other parameters in HandBreak, to better demonste above scenario.(eg. Frame rate)

To log the execution time of a bash script you can use the linux "time" utility.

```
$ time < command/bash script >
```

2. Performance comparison exercise with Hadoop (40%)

Follow the instructions in Appendix A to install hadoop on one Linux virtual machine. Then try out “wordcount” and “grep” example codes that are provided with the hadoop installation.

You can use following documents as input (Choose Plain Text version).

- The Outline of Science, Vol. 1 (of 4) by J. Arthur Thomson
- The Notebooks of Leonardo Da Vinci
- Ulysses by James Joyce
- The Art of War by 6th cent. B.C. Sunzi

- The Adventures of Sherlock Holmes by Sir Arthur Conan Doyle
- The Devil's Dictionary by Ambrose Bierce
- Encyclopaedia Britannica, 11th Edition, Volume 4, Part 3

Turn the single node into a multi node setup (with 2 VMs, one as Slave and Master and the other as just Slave). Then perform following experiments.

a) Measure and present completion times for both wordcount and grep applications with various inputs. Compare how it varies in single and multi node setup.

b) Show the differences in the input and output data, of mapper and reducer. Discuss how data changes during mapper and reducer processes.

Hint: Disable the reducer so to see data gone through just mapper.

Exercises for extra marks (30%)

a) Sort the input files before parsing to mapper. Show and compare the results to show the differences with sorting and without sorting. Discuss why you observe such behaviour. (15%)

b) Try changing the amount of mapper and reducer processes before running wordcount or/and grep examples. Discuss and show how these numbers change performance results evidently. (15%)

Submission

Students need to submit to the coursework submission system the following documents:

1) A report (better in editable .doc format). The cover page of the report shall have your name and registration number.

Appendix A

Install Hadoop.

Add dedicated hadoop system user.

```
$ sudo addgroup hadoop
$ sudo adduser --ingroup hadoop hduser
$ sudo adduser hduser sudo
$ su hduser
```

Download Hadoop 2.x to "/home/hduser/yarn" folder from here. Extract it to a folder. Then Set permissions.

```
$ cd ~
$ sudo chown -R hduser:hadoop hadoop-2.6.0
```

Install Java JDK

```
$ sudo apt-get install python-software-properties
$ sudo add-apt-repository ppa:webupd8team/java
$ sudo apt-get update
$ sudo apt-get install oracle-java7-installer
```

Install and configure SSH server

```
$ sudo apt-get install openssh-server
$ ssh-keygen -t rsa -P ""
$ cat $HOME/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys
```

Set environment variables.

Add following lines to ~/.bashrc.

```
$ export HADOOP_HOME=$HOME/yarn/hadoop-2.6.0
$ export HADOOP_MAPRED_HOME=$HOME/yarn/hadoop-2.6.0
$ export HADOOP_COMMON_HOME=$HOME/yarn/hadoop-2.6.0
$ export HADOOP_HDFS_HOME=$HOME/yarn/hadoop-2.6.0
$ export HADOOP_YARN_HOME=$HOME/yarn/hadoop-2.6.0
$ export HADOOP_CONF_DIR=$HOME/yarn/hadoop-2.6.0/etc/hadoop
```

Then source the bashrc file.

```
$ source ~/.bashrc
```

Create Directories

```
$ mkdir -p $HOME/yarn/yarn_data/hdfs/namenode
$ mkdir -p $HOME/yarn/yarn_data/hdfs/datanode
```

Edit configuration files.

```
$ cd $HADOOP_YARN_HOME
```

Add the following properties under the configuration tag in the following files.

etc/hadoop/yarn-site.xml:

```
<property>
   <name>yarn.nodemanager.aux-services</name>
   <value>mapreduce_shuffle</value>
</property>
<property>
   <name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>
   <value>org.apache.hadoop.mapred.ShuffleHandler</value>
</property>
```

etc/hadoop/core-site.xml:

```
<property>
   <name>fs.default.name</name>
   <value>hdfs://localhost:9000</value>
</property>
```

etc/hadoop/hdfs-site.xml:

```
<property>
   <name>dfs.replication</name>
   <value>1</value>
 </property>
 <property>
   <name>dfs.namenode.name.dir</name>
   <value>file:/home/hduser/yarn/yarn_data/hdfs/namenode</value>
 </property>
 <property>
   <name>dfs.datanode.data.dir</name>
   <value>file:/home/hduser/yarn/yarn_data/hdfs/datanode</value>
 </property>
```

etc/hadoop/mapred-site.xml:

```
<?xml version="1.0"?>
<configuration>
   <property>
      <name>mapreduce.framework.name</name>
      <value>yarn</value>
   </property>
</configuration>
```

Format namenode

```
$ bin/hadoop namenode –format
```

Start Hadoop File System Processes

```
$ sbin/hadoop-daemon.sh start namenode
$ sbin/hadoop-daemon.sh start datanode
$ jps
```

Start Hadoop Map-Reduce Processes.

```
$ sbin/yarn-daemon.sh start resourcemanager
$ sbin/yarn-daemon.sh start nodemanager
$ sbin/mr-jobhistory-daemon.sh start historyserver
$ jps
```

Web interfaces:

http://localhost:50070

http://localhost:8088
http://master:19888/jobhistory

Done!