HADOOP CLUSTER SETUP GUIDE:

Passwordless SSH Sessions:

Before we start our installation, we have to ensure that passwordless SSH Login is possible to any of the Linux machines of CS120. In order to do so, log in to your account and run the following commands:
[For Further understanding, look up:
https://hadoopabcd.wordpress.com/2015/05/16/linux-password-less-ssh-logins/]

```
ssh-keygen -t rsa -P "" -f ~/.ssh/id_rsa
cat $HOME/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys
```

Setting up Config Files:

Step 1. Create a new folder for your configuration files:

```
mkdir ~/hadoopConf
```

Step 2. Create core-site.xml, hdfs-site.xml, mapred-site.xml, and yarn-site.xml. Use below commands (separately) to download and unpack the sample files into ~/hadoopConf

```
 conf.tar - https://colostate.instructure.com/files/22371813/download?download_frd=1
tar xvf conf.tar -C ~/hadoopConf
```

For the files core-site.xml, mapred-site.xml and hdfs-site.xml, **anything with “HOST” or “PORT” in the config files should be replaced with actual hostnames and** port numbers assigned to you.
Make sure that you select 5-10 nodes (CS120 machines only) for your slaves file that is detailed below in Step 4.
You need to select a set of available ports from the non-privileged port range. Do not select any ports between 56,000 and 57,000. These are dedicated for the shared HDFS cluster. To reduce possible port conflicts, you will be each assigned a port range

Step 3. Set the environment variables in your .bashrc file

```
export HADOOP_HOME="/usr/local/hadoop/3.1.2"
export HADOOP_COMMON_HOME="${HADOOP_HOME}"
export HADOOP_HDFS_HOME="${HADOOP_HOME}"
export HADOOP_MAPRED_HOME="${HADOOP_HOME}"
export YARN_HOME="${HADOOP_HOME}"
export HADOOP_CONF_DIR=<path to your config directory mentioned in Step 2>
export YARN_CONF_DIR="${HADOOP_CONF_DIR}"
export HADOOP_LOG_DIR="/tmp/${USER}/hadoop-logs"
export YARN_LOG_DIR="/tmp/${USER}/yarn-logs"
export JAVA_HOME="/usr/lib/jvm/java-1.8.0-openjdk"
export HADOOP_OPTS="-Dhadoop.tmp.dir=/s/${HOSTNAME}/a/tmp/hadoop-${USER}"
export HADOOP_CLASSPATH="${JAVA_HOME}/lib/tools.jar"
```

For example, if you created the hadoopConf in your home directory (as mentioned in Step 1), HADOOP_CONF_DIR can be specified as;

```
export HADOOP_CONF_DIR="${HOME}/hadoopConf"
```

Step 5. After setting up above environment variables, to apply the changes, run

```
source ~/.bashrc
```

Step 6. Modify masters and workers file inside the hadoopConf folder.

Masters file should contain the name of the machine you choose as your secondary namenode.

Slaves contain the list of all your worker nodes.

Programming assignment 1 requires at least 5 such nodes. (there should not be overlapping nodes in these 2 files)

Running HDFS

a. First Time Configuration

Step 1. Log into your namenode

You will have specified your namenode in core-site.xml.

```
<property>
  <name>fs.default.name</name>
  <value>hdfs://host:port</value>
</property>
```

Step 2. Format your namenode.

```
$HADOOP_HOME/bin/hdfs namenode -format
```

b. **Starting HDFS**

Run the start script:

```
$HADOOP_HOME/sbin/start-dfs.sh
```

Step 3. Check the web portal to make sure that HDFS is running, open on your browser the url **http://<namenode>:<port given>** for the property **dfs.namenode.http-address** in hdfs-site.xml

(Please read https://colostate.instructure.com/files/22375155/download?download_frd=1 to learn more about opening web user interfaces remotely)

c. Stopping HDFS

Step 1. Log into your namenode

Step 2. Run the stop script:

```
$HADOOP_HOME/sbin/stop-dfs.sh
```

Running Yarn

a**. Starting Yarn**

Step 1. Log into your resource manager
(Resource Manager is the node you set as HOST
in the yarn-site.xml)

Step 2. Run the start script

```
$HADOOP_HOME/sbin/start-yarn.sh
```

Step 3. Check the web portal to make sure yarn is running
http://<resourcemanager>:<port given for the property yarn.resourcemanager.webapp.address>

(Please read https://colostate.instructure.com/files/22375155/download?download_frd=1 to learn more about opening web user interfaces remotely)

b. **Stopping Yarn**

Step 1. Log into your resource manager
Step 2. Run the stop script

```
$HADOOP_HOME/sbin/stop-yarn.sh
```

Running a Job Locally

a. Setup

Step 1. Open `mapred-site.xml`

Step 2. Change the value of property `mapreduce.framework.name` to `local`.

To be safe, restart HDFS and ensure yarn is stopped

b. Running the Job From any node:

```
$HADOOP_HOME/bin/hadoop jar <path_to_your.jar> <yourMainClass>
<argument_1>...<argument_N>
```

Running a Job in Yarn

a. Setup

Step 1. Open mapred-site.xml

Step 2. Change the value of property `mapreduce.framework.name` to `yarn`

To be safe, restart HDFS and yarn

b. Running the Job From any node

```
$HADOOP_HOME/bin/hadoop jar <path_to_your.jar> <yourMainClass>
<argument_1>...<argument_N>
```