**Lab 4A: Run Spark on a single computing node**

(1) Copy, unzip, and expand spark-2.3.1-bin-hadoop2.7.tgz.

**[11:06:35] mfukuda@cssmpi1h: ~ $** cp ~css534/spark-2.3.1-bin-hadoop2.7.tgz ~

**[11:07:09] mfukuda@cssmpi1h: ~ $** gzip -d spark-2.3.1-bin-hadoop2.7.tgz

**[11:11:29] mfukuda@cssmpi1h: ~ $** tar -xvf - < spark-2.3.1-bin-hadoop2.7.tar

(2) Update your .bash\_profile to run Spark.

**[11:15:12] mfukuda@cssmpi1h: ~ $** nano .bash\_profile

export JAVA\_HOME=/usr/lib/jvm/java-1.8.0

export SPARK\_HOME=/home/NETID/mfukuda/spark-2.3.1-bin-hadoop2.7

export PATH=$SPARK\_HOME/bin:$PATH

Plesae note that Spark 2.3.1 works only on Java 1.8. Spark 3.0 runs with Java 11.

**[11:15:12] mfukuda@cssmpi1h: ~ $** source .bash\_profile

(3) Run pySpark.

**[13:06:16] mfukuda@cssmpi1h: ~ $** cd spark-2.3.1-bin-hadoop2.7/bin

**[13:06:25] mfukuda@cssmpi1h: bin $** pyspark

Python 2.7.5 (default, Oct 30 2018, 23:45:53)

[GCC 4.8.5 20150623 (Red Hat 4.8.5-36)] on linux2

Type "help", "copyright", "credits" or "license" for more information.

2019-11-12 13:06:46 WARN NativeCodeLoader:62 - Unable to load native-hadoop library for your platform... using builtin-java classes where applicable

Setting default log level to "WARN".

To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).

Welcome to

\_\_\_\_ \_\_

/ \_\_/\_\_ \_\_\_ \_\_\_\_\_/ /\_\_

\_\ \/ \_ \/ \_ `/ \_\_/ '\_/

/\_\_ / .\_\_/\\_,\_/\_/ /\_/\\_\ version 2.3.1

/\_/

Using Python version 2.7.5 (default, Oct 30 2018 23:45:53)

SparkSession available as 'spark'.

>>> document = sc.textFile("/home/NETID/css534/programming/Spark/sample.txt")

>>> len( document.flatMap( lambda s: s.split( ) ).collect( ) )

11

>>> numbers = sc.parallelize( [0, 1, 2, 3, 4, 5] )

>>> numbers.max( )

5

>>> quit( )

(3) Compile a Spark program.

You may copy Spark sample programs under ~css534/programming/Spark/. Note that you can't copy MyClass.java, its jar and classes; JavaWordCountFlatMap2.java, its jar and classes; as well as ShortestPath.java, its jar and class, because they are Lab4a-b and Program4's key answers.

**[11:47:20] mfukuda@cssmpi1h: ~ $** cd

**[11:47:20] mfukuda@cssmpi1h: ~ $** cp -r ~css534/programming/Spark .

**[11:47:41] mfukuda@cssmpi1h: ~ $** cd Spark

If you look at the jars/ directory, this is a copy of ~/spark-2.3.1-bin-hadoop2.7. To compile typical sample programs such as WordCount, PageRank, and ShortestPath, you need the following dependencies:

spark-core\_2.11-2.3.1.jar: JavaSparkContext, JavaRDD, JavaPairRDD

spark-sql\_2.11-2.3.1.jar: SparkSession

spark-liabrary\_2.11.8.jar: SparkSession.builder().scala.\*, Tuple2

google-collections-1.0.jar: PageRank

This means that javac needs to point them with -cp.

/usr/lib/jvm/java-1.8.0/bin/javac -cp jars/spark-core\_2.11-2.3.1.jar:jars/spark-sql\_2.11-2.3.1.jar:jars/scala-library-2.11.8.jar:google-collections-1.0.jar **YourProgram.java**

Make sure that you are using java-1.8.0’s java compiler.

To run your program, you need to archive all the class files into a jar file.

/usr/bin/jvm/java-1.8.0/bin/jar -cvf YourProgram.jar YourClass1.class YourClass2.class ...

Make sure that you are using java-1.8.0’s java archiver.

For your convenience, you may use compile.sh to compile and create a jar file of your Spark program.

**[12:57:00] mfukuda@cssmpi1h: Spark $** ./compile.sh MyClass

added manifest

adding: MyClass.class(in = 3341) (out= 1602)(deflated 52%)

adding: MyClassMax.class(in = 628) (out= 356)(deflated 43%)

adding: MyClass$MyClassMax.class(in = 693) (out= 385)(deflated 44%)

(4) Run a Spark program.

To run your program, simply run with spark-submit

spark-submit --class MyClass --master local MyClass.jar

For your comvenience, you may use run.sh to run your program.

**[12:55:52] mfukuda@cssmpi1h: ~/css534/programming/Spark $** ./run.sh MyClass

**Lab 4B: Run Spark over multiple computing nodes (in SparkCluster).**

(5) Configure spark-2.3.1-bin-hadoop2.7/conf/spark-env.

**[13:13:02] mfukuda@cssmpi1h: ~ $** cd ~/spark-2.3.1-bin-hadoop2.7

**[13:13:02] mfukuda@cssmpi1h: spark-2.3.1-bin-hadoop2.7 $** cd conf

**[13:13:23] mfukuda@cssmpi1h: conf $** cp spark-env.sh.template spark-env.sh

**[13:13:45] mfukuda@cssmpih1: conf $** nano spark-env.sh

# Options for the daemons used in the standalone deploy mode

SPARK\_MASTER\_HOST=cssmpi1h.uwb.edu

SPARK\_MASTER\_PORT=58000

SPARK\_MASTER\_WEBUI\_PORT=58001

JAVA\_HOME=/usr/lib/jvm/java-1.8.0

# - SPARK\_MASTER\_OPTS, to set config properties only for the master (e.g. "-Dx=y")

# - SPARK\_WORKER\_CORES, to set the number of cores to use on this machine

# - SPARK\_WORKER\_MEMORY, to set how much total memory workers have to give executors (e.g. 1000m, 2g)

# - SPARK\_WORKER\_PORT the default is random

SPARK\_WORKER\_WEBUI\_PORT=58002

(6) Configure slaves.

**[13:22:53] mfukuda@cssmpi1h: conf $** cp slaves.template slaves

**[13:25:29] mfukuda@cssmpi1h: conf $** nano slaves

# A Spark Worker will be started on each of the machines listed below.

**cssmpi1.uwb.edu**

**cssmpi2h.uwb.edu**

**cssmpi3h.uwb.edu**

**cssmpi4h.uwb.edu**

(7) As far as you successfully completed HW2 MPI, definitely skip step 7. Otherwise, make sure that SSH key agents are running.

ssh-keygen -t dsa -f ~/.ssh/id\_dsa

cat ~/.ssh/id\_dsa.pub >> ~/.ssh/authorized\_keys

Just in case, try to login these machines to make sure that you won’t receive any yes/no or login questions from the system.

ssh cssmpi1h.uwb.edu

ssh cssmpi2h.uwb.edu

ssh cssmpi3h.uwb.edu

ssh cssmpi4h.uwb.edu

(8) Start your Spark cluster.

**[13:32:24] mfukuda@cssmpi1h: conf $** cd ~/spark-2.3.1-bin-hadoop2.7/sbin

**[13:32:44] mfukuda@cssmpi1h: sbin $** ./start-master.sh

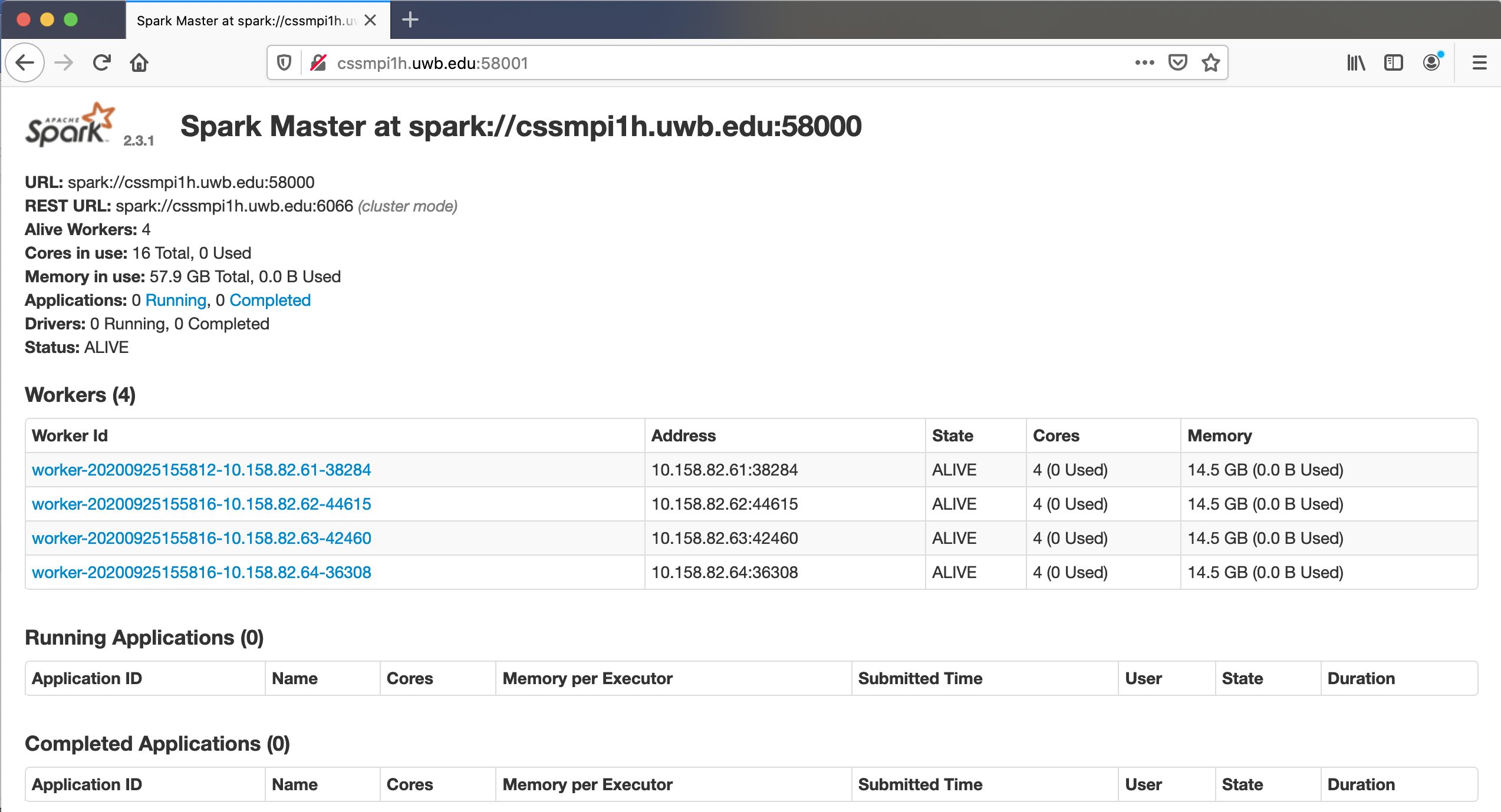
**[14:01:57] mfukuda@cssmpi1h: sbin $** ./start-slaves.sh

(8) Check if your Spark cluster.

Open a web browser and type in URL

http://cssmpi1h.uwb.edu:58001/

Note that the port is yours but not 58001.



(9) Submit your Spark program.

**[14:01:57] mfukuda@cssmpi1h: sbin $** cd ~/Spark

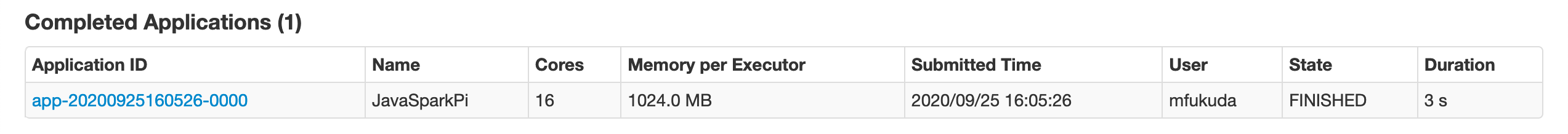
**[14:07:30] mfukuda@cssmpi1h: Spark $** spark-submit --class JavaSparkPi --master spark://cssmpi1h.uwb.edu:58000 JavaSparkPi.jar 10

2019-11-12 14:07:40 WARN NativeCodeLoader:62 - Unable to load native-hadoop library for your platform... using builtin-java classes where applicable

2019-11-12 14:07:40 INFO SparkContext:54 - Running Spark version 2.3.1

2019-11-12 14:07:40 INFO SparkContext:54 - Submitted application: JavaSparkPi

Pi is roughly 3.143516



(10) Stop your Spark cluster

**[14:10:23] mfukuda@cssmpi1: Spark$** cd ~/spark-2.3.1-bin-hadoop2.7/sbin

**[14:10:34] mfukuda@cssmpi1: sbin $** ./stop-all.sh

cssmpi3h.uwb.edu: stopping org.apache.spark.deploy.worker.Worker

cssmpi4h.uwb.edu: stopping org.apache.spark.deploy.worker.Worker

cssmpi2h.uwb.edu: stopping org.apache.spark.deploy.worker.Worker

cssmpi1h.uwb.edu: stopping org.apache.spark.deploy.worker.Worker

stopping org.apache.spark.deploy.master.Master