Qinzhi Peng , qinzhip

cluster user ID : student159

cluster password: Pqz!0550119

Part 1
Task0
// copy java file from public to local
cp /home/public/WordCount.java /home/student159/Project5/Part_1/Task0

// copy words.txt to HDFS
hadoop dfs -copyFromLocal /home/public/words.txt /user/student159/input

// compile java file
javac -classpath /usr/local/hadoop/hadoop-core-1.2.1.jar:./home/student159/Project5/Part_1/Task0/wordcount_classes -d /home/student159/wordcount_classes WordCount.java

// generate a jar file
jar -cvf wordcount.jar -C /home/student159/wordcount_classes .

// deploy the jar file and test it against /home/public/words.txt
hadoop jar /home/student159/Project5/Part_1/Task0/wordcount.jar org.myorg.WordCount /user/student159/input/words.txt /user/student159/output/task0_output/

// examine the outputs of the three reducers
hadoop dfs -cat /user/student159/output/task0_output/part-r-00000
hadoop dfs -cat /user/student159/output/task0_output/part-r-00001
hadoop dfs -cat /user/student159/output/task0_output/part-r-00002

// place the results in the output folder
hadoop dfs -getmerge /user/student159/output /home/student159/Project5/Part_1/Task0/Task0Output/

```
[student159@heinz-jumbo Task0Output]$ cat /home/student159/Project5/Part_1/Task0/Task0Output/output
A       1
Aani    1
Aaron   1
Aaronite        1
Aaronitic       1
Ababdeh 1
Abanic  1
Abantes 1
Abarambo        1
Abasgi  1
Abassin 1
Abbie   1
Abdiel  1
Abelia  1
Abelian 1
Abelicea        1
Abencerrages    1
Aberia  1
Abhorson        1
Abietineae      1
Abigail 1
Abipon  1
Abkhas  1
Abner   1
Abobra  1
Abongo  1
Abraham 1
Abrahamic       1
Abrahamite      1
Abrahamitic     1
Abrocoma        1
Abrus   1
Absaroka        1
Absi    1
Absyrtus        1
Abutilon        1
Acacia  1
Acacian 1
Academus        1
Acalypha        1
Acalyptrata     1
Acanthaceae     1
Acantharia      1
Acanthocereus   1
Acanthodidae    1
Acanthodini     1
```

Task1

// transfer java file from local machine to cluster

scp /Users/pengqinzhi/Desktop/Distributed-Systems/Project/5/Project5Part1/src/org/myorg/LetterCounter.java student159@heinz-jumbo.heinz.cmu.local:/home/student159/Project5/Part_1/Task1

// compile java file

mkdir classes

javac -classpath /usr/local/hadoop/hadoop-core-1.2.1.jar:./home/student159/Project5/Part_1/Task1/classes -d /home/student159/Project5/Part_1/Task1/classes LetterCounter.java

// generate a jar file

jar -cvf lettercount.jar -C /home/student159/Project5/Part_1/Task1/classes .

// deploy the jar file and test it against /home/public/words.txt

hadoop jar /home/student159/Project5/Part_1/Task1/lettercount.jar org.myorg.LetterCounter /user/student159/input/words.txt /user/student159/output/task1_output

// check

hadoop dfs -ls /user/student159/output/task1_output

//delete

hadoop dfs -rmr /user/student159/output/task1_output

// examine the outputs of the three reducers
hadoop dfs -cat /user/student159/output/task1_output/part-r-00000
hadoop dfs -cat /user/student159/output/task1_output/part-r-00001
hadoop dfs -cat /user/student159/output/task1_output/part-r-00002

// place the results in the output folder
hadoop dfs -getmerge /user/student159/output/task1_output/
/home/student159/Project5/Part_1/Task1/Task1Output/

// sort
sort -k 2nr task1_output

```
[student159@heinz-jumbo Task0Output]$ cat /home/student159/Project5/Part_1/Task1/Task1Output/task1_output
A       2559
D       950
G       1018
J       484
M       1907
P       2291
S       2403
V       361
Y       139
b       39051
e       234456
h       63230
k       15619
n       158086
q       3657
t       151270
w       13531
z       8232
B       1395
E       918
H       1138
K       546
N       682
Q       77
T       1578
W       337
Z       230
c       100953
f       23689
i       200573
l       129403
o       170081
r       160292
u       87172
x       6840
C       2496
F       478
I       496
L       1073
O       630
R       716
U       208
X       92
a       197019
d       67252
g       46085
j       2685
m       68804
p       75873
s       137157
v       19819
y       51544
```

Task2
// compile java file
mkdir classes
javac -classpath /usr/local/hadoop/hadoop-core-1.2.1.jar:./home/student159/Project5/Part_1/Task2/classes -d /home/student159/Project5/Part_1/Task2/classes WordSearch.java

// generate a jar file
jar -cvf wordsearch.jar -C /home/student159/Project5/Part_1/Task2/classes .

// deploy the jar file and test it against /home/public/words.txt
hadoop dfs -rmr /user/student159/output/task2_output
hadoop jar /home/student159/Project5/Part_1/Task2/wordsearch.jar org.myorg.WordSearch /user/student159/input/words.txt /user/student159/output/task2_output

// examine the outputs of the three reducers
hadoop dfs -cat /user/student159/output/task2_output/part-r-00000
hadoop dfs -cat /user/student159/output/task2_output/part-r-00001
hadoop dfs -cat /user/student159/output/task2_output/part-r-00002

// place the results in the output folder
hadoop dfs -getmerge /user/student159/output/task2_output/ /home/student159/Project5/Part_1/Task2/Task2Output/

```
[student159@heinz-jumbo Task0Output]$ cat /home/student159/Project5/Part_1/Task2/Task2Output/task2_output
artifact
artifactitious
benefactive
benefactor
bilifaction
calefactory
chylifactive
cofactor
cretefaction
dissatisfactoriness
dissatisfactory
factful
facticide
factional
factionary
factiousness
factish
factitial
factitively
factive
factorist
factorization
factotum
factrix
factualness
factum
insatisfaction
labefactation
liquefaction
lithifaction
malefactory
manufactory
manufacturess
metallifacture
nitrifaction
nonfactious
nonmanufactured
nonolfactory
olfactology
olfactometric
olfactor
olfactory
overfactiousness
petrifactive
pinguefaction
postfact
predissatisfaction
prefactory
putrefactible
putrifacted
rarefactive
rubefaction
```

Task3
// copy java file from public to local
cp /home/public/MaxTemperature.java /home/student159/Project5/Part_1/Task3
cp /home/public/MaxTemperatureMapper.java /home/student159/Project5/Part_1/Task3
cp /home/public/MaxTemperatureReducer.java /home/student159/Project5/Part_1/Task3

// copy combinedYears.txt to HDFS
hadoop dfs -copyFromLocal /home/public/combinedYears.txt /user/student159/input

// compile three Java classes using a library of Hadoop classes
javac -classpath /usr/local/hadoop/hadoop-core-1.2.1.jar:./temperature_classes -d temperature_classes MaxTemperatureMapper.java

javac -classpath /usr/local/hadoop/hadoop-core-1.2.1.jar:./temperature_classes -d temperature_classes MaxTemperatureReducer.java

javac -classpath /usr/local/hadoop/hadoop-core-1.2.1.jar:./temperature_classes -d temperature_classes MaxTemperature.java

// create a new jar file
jar -cvf temperature.jar -C temperature_classes/ .

// deploy the jar file and test it against the data set under /home/public/combinedYears.txt
hadoop jar /home/student159/Project5/Part_1/Task3/temperature.jar edu.cmu.andrew.mm6.MaxTemperature /user/student159/input/combinedYears.txt /user/student159/output/task3_output

// place the results in the output folder
mkdir /home/student159/Project5/Part_1/Task3/Task3Output/
hadoop dfs -getmerge /user/student159/output/task3_output/ /home/student159/Project5/Part_1/Task3/Task3Output/

```
[student159@heinz-jumbo Task0Output]$ cat /home/student159/Project5/Part_1/Task3/Task3Output/task3_output
1901    317
1902    244
```

Task4
// transfer a directory from local to cluster
scp -r /Users/pengqinzhi/Desktop/Distributed-Systems/Project/5/Project5Part1/src/edu/cmu/andrew/student159 student159@heinz-jumbo.heinz.cmu.local:/home/student159/Project5/Part_1/Task4

// ompile three Java classes using a library of Hadoop classes
mkdir mintemperature_classes

javac -classpath /usr/local/hadoop/hadoop-core-1.2.1.jar:./mintemperature_classes -d mintemperature_classes MinTemperatureMapper.java

javac -classpath /usr/local/hadoop/hadoop-core-1.2.1.jar:./mintemperature_classes -d mintemperature_classes MinTemperatureReducer.java

javac -classpath /usr/local/hadoop/hadoop-core-1.2.1.jar:./mintemperature_classes -d mintemperature_classes MinTemperature.java

// create a new jar file
jar -cvf mintemperature.jar -C mintemperature_classes/ .

// deploy the jar file and test it against the data set under /home/public/combinedYears.txt
hadoop dfs -rmr /user/student159/output/task4_output
hadoop jar /home/student159/Project5/Part_1/Task4/mintemperature.jar edu.cmu.andrew.student159.MinTemperature /user/student159/input/combinedYears.txt /user/student159/output/task4_output

// place the results in the output folder
mkdir /home/student159/Project5/Part_1/Task4/Task4Output/
hadoop dfs -getmerge /user/student159/output/task4_output/ /home/student159/Project5/Part_1/Task4/Task4Output/

```
[student159@heinz-jumbo Task0Output]$ cat /home/student159/Project5/Part_1/Task4/Task4Output/task4_output
1901    -333
1902    -328
```

Task5
// transfer
scp /Users/pengqinzhi/Desktop/Distributed-Systems/Project/5/Project5Part1/src/edu/cmu/andrew/student159/RapesPlusRobberies.java student159@heinz-jumbo.heinz.cmu.local:/home/student159/Project5/Part_1/Task5

// copy
hadoop dfs -copyFromLocal /home/public/P1V.txt /user/student159/input

// compile three Java classes using a library of Hadoop classes
mkdir classes
javac -classpath /usr/local/hadoop/hadoop-core-1.2.1.jar:./classes -d classes RapesPlusRobberies.java

// create a new jar file
jar -cvf rapesplusrobberies.jar -C classes/ .

// deploy the jar file and test it against the data set under /home/public/P1V.txt
hadoop dfs -rmr /user/student159/output/task5_output

hadoop jar /home/student159/Project5/Part_1/Task5/rapesplusrobberies.jar
edu.cmu.andrew.student159.RapesPlusRobberies /user/student159/input/P1V.txt
/user/student159/output/task5_output

// examine the outputs of the three reducers
hadoop dfs -cat /user/student159/output/task5_output/part-r-00000
hadoop dfs -cat /user/student159/output/task5_output/part-r-00001
hadoop dfs -cat /user/student159/output/task5_output/part-r-00002

// place the results in the output folder
mkdir /home/student159/Project5/Part_1/Task5/Task5Output/
hadoop dfs -getmerge /user/student159/output/task5_output/
/home/student159/Project5/Part_1/Task5/Task5Output/

```
[student159@heinz-jumbo Task0Output]$ cat /home/student159/Project5/Part_1/Task5/Task5Output/task5_output
count   19283
```

Task 6
// transfer
scp /Users/pengqinzhi/Desktop/Distributed-
Systems/Project/5/Project5Part1/src/edu/cmu/andrew/student159/OaklandCrimeStats.java
student159@heinz-jumbo.heinz.cmu.local:/home/student159/Project5/Part_1/Task6

// compile three Java classes using a library of Hadoop classes
mkdir classes
javac -classpath /usr/local/hadoop/hadoop-core-1.2.1.jar:./classes -d classes
OaklandCrimeStats.java

// create a new jar file
jar -cvf oaklandcrimestats.jar -C classes/ .

// deploy the jar file and test it against the data set under /home/public/P1V.txt
hadoop dfs -rmr /user/student159/output/task6_output
hadoop jar /home/student159/Project5/Part_1/Task6/oaklandcrimestats.jar
edu.cmu.andrew.student159.OaklandCrimeStats /user/student159/input/P1V.txt
/user/student159/output/task6_output

// examine the outputs of the three reducers
hadoop dfs -cat /user/student159/output/task6_output/part-r-00000
hadoop dfs -cat /user/student159/output/task6_output/part-r-00001
hadoop dfs -cat /user/student159/output/task6_output/part-r-00002

// place the results in the output folder
mkdir /home/student159/Project5/Part_1/Task6/Task6Output/
hadoop dfs -getmerge /user/student159/output/task6_output/
/home/student159/Project5/Part_1/Task6/Task6Output/

```
[student159@heinz-jumbo Task0Output]$ cat /home/student159/Project5/Part_1/Task6/Task6Output/task6_output
count   58
```

Task7
// copy from public
hadoop dfs -copyFromLocal /home/public/CrimeLatLonXYTabs.txt /user/student159/input

// transfer file
scp /Users/pengqinzhi/Desktop/Distributed-Systems/Project/5/Project5Part1/src/edu/cmu/andrew/student159/OaklandCrimeStatsKml.java student159@heinz-jumbo.heinz.cmu.local:/home/student159/Project5/Part_1/Task7

// compile three Java classes using a library of Hadoop classes
mkdir classes
javac -classpath /usr/local/hadoop/hadoop-core-1.2.1.jar:./classes -d classes OaklandCrimeStatsKml.java

// create a new jar file
jar -cvf oaklandcrimestatskml.jar -C classes/ .

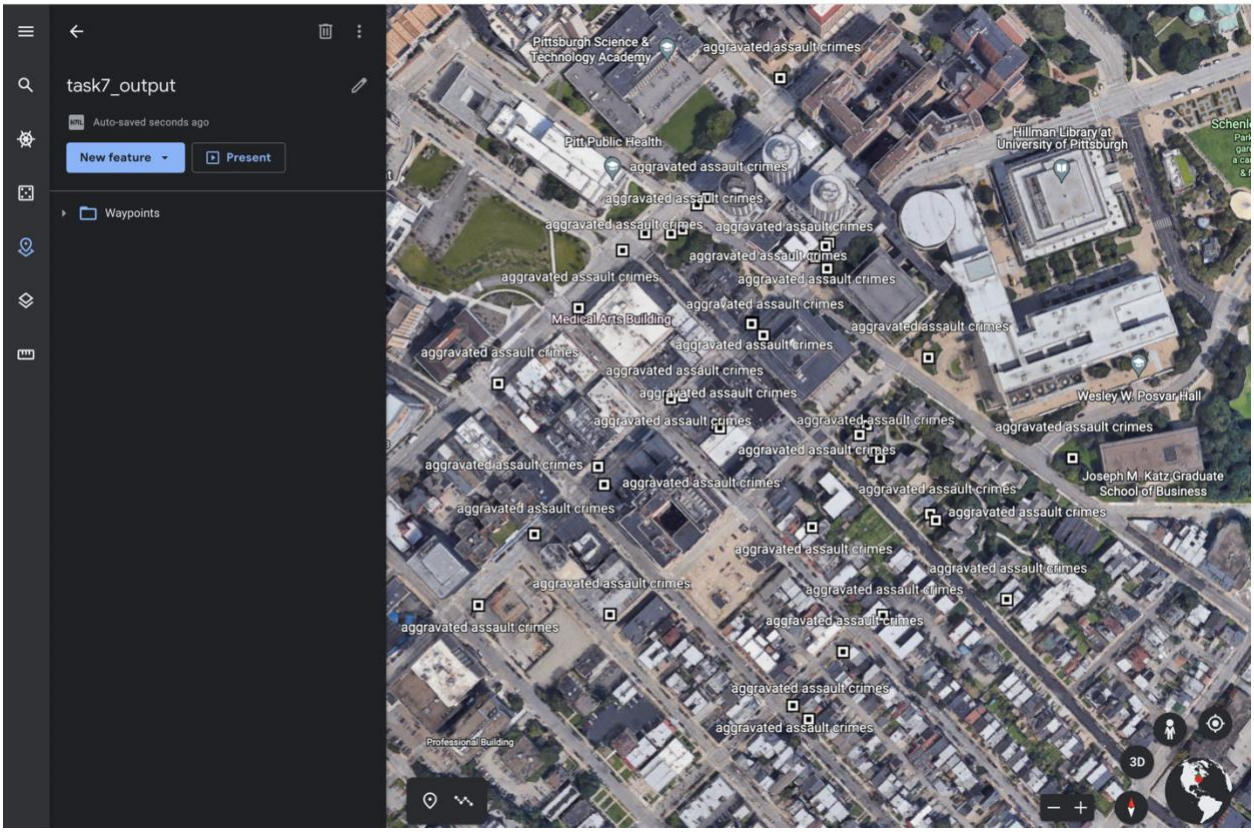
// deploy the jar file and test it against the data set under CrimeLatLonXYTabs.txt
hadoop dfs -rmr /user/student159/output/task7_output
hadoop jar /home/student159/Project5/Part_1/Task7/oaklandcrimestatskml.jar edu.cmu.andrew.student159.OaklandCrimeStatsKml /user/student159/input/CrimeLatLonXYTabs.txt /user/student159/output/task7_output

// examine the outputs of the three reducers
hadoop dfs -cat /user/student159/output/task7_output/part-r-00000
hadoop dfs -cat /user/student159/output/task7_output/part-r-00001
hadoop dfs -cat /user/student159/output/task7_output/part-r-00002

// place the results in the output folder
mkdir /home/student159/Project5/Part_1/Task7/Task7Output/
hadoop dfs -getmerge /user/student159/output/task7_output/ /home/student159/Project5/Part_1/Task7/Task7Output/

// retrieve the output
scp student159@heinz-jumbo.heinz.cmu.local:/home/student159/Project5/Part_1/Task7/Task7Output/task7_output /Users/pengqinzhi/Desktop

```
[student159@heinz-jumbo Task0Output]$ cat /home/student159/Project5/Part_1/Task7/Task7Output/task7_output
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://www.opengis.net/kml/2.2">
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.95773576,40.44231637
 </coordinates></Point></Placemark>
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.95871096,40.44031908
 </coordinates></Point></Placemark>
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.9561512,40.44208775
 </coordinates></Point></Placemark>
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.95809936,40.44065047
 </coordinates></Point></Placemark>
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.95398897,40.44082613
 </coordinates></Point></Placemark>
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.95708769,40.44102886
 </coordinates></Point></Placemark>
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.95519815,40.44041241
 </coordinates></Point></Placemark>
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.95526271,40.4414964
 </coordinates></Point></Placemark>
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.95670703,40.44164308
 </coordinates></Point></Placemark>
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.9561512,40.44208775
 </coordinates></Point></Placemark>
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.95457963,40.43988667
 </coordinates></Point></Placemark>
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.95568695,40.44082789
 </coordinates></Point></Placemark>
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.95901234,40.44132082
 </coordinates></Point></Placemark>
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.95728335,40.44250806
 </coordinates></Point></Placemark>
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.95809936,40.44065047
 </coordinates></Point></Placemark>
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.95708769,40.44102886
 </coordinates></Point></Placemark>
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.9575101,40.44231467
 </coordinates></Point></Placemark>
 <Placemark><name>aggravated assault crimes</name><Point><coordinates>
-79.9575148,40.44122178
```



Part2
Task0
Using the count method of the JavaRDD class, display the number of lines in "The Tempest".

```
// configure spark
SparkConf sparkConf = new SparkConf().setMaster("local").setAppName("JD
Tempest Analytics");

// create a JavaSparkContext that loads settings from system properties
JavaSparkContext sparkContext = new JavaSparkContext(sparkConf);

// read an input text file to RDD
JavaRDD<String> inputFile = sparkContext.textFile(fileName);

// split each line
JavaRDD<String> linesFromFile = inputFile.flatMap(content ->
Arrays.asList(content.split("\n")));

// count number of lines
long countLines = linesFromFile.count();
System.out.println("Number of lines: " + countLines);
```

```
/Users/pengqinzhi/Library/Java/JavaVirtualMachines/liberica-1.8.0_322/bin/java ...
22/04/22 22:45:26 INFO spark.SecurityManager: Changing view acls to: pengqinzhi
22/04/22 22:45:26 INFO spark.SecurityManager: SecurityManager: authentication disabled; ui acls disabled; users with view permissions: Set(pengqinzhi)
22/04/22 22:45:26 INFO slf4j.Slf4jLogger: Slf4jLogger started
22/04/22 22:45:26 INFO Remoting: Starting remoting
22/04/22 22:45:27 INFO Remoting: Remoting started; listening on addresses :[akka.tcp://spark@192.168.1.180:51640]
22/04/22 22:45:27 INFO Remoting: Remoting now listens on addresses: [akka.tcp://spark@192.168.1.180:51640]
22/04/22 22:45:27 INFO spark.SparkEnv: Registering MapOutputTracker
22/04/22 22:45:27 INFO spark.SparkEnv: Registering BlockManagerMaster
22/04/22 22:45:27 INFO storage.DiskBlockManager: Created local directory at /var/folders/ts/q9gpv9j931jgspdt_hzq6w940000gn/T/spark-local-20220422224527-3b03
22/04/22 22:45:27 INFO storage.MemoryStore: MemoryStore started with capacity 4.3 GB.
22/04/22 22:45:27 INFO network.ConnectionManager: Bound socket to port 51641 with id = ConnectionManagerId(192.168.1.180,51641)
22/04/22 22:45:27 INFO storage.BlockManagerMaster: Trying to register BlockManager
22/04/22 22:45:27 INFO storage.BlockManagerInfo: Registering block manager 192.168.1.180:51641 with 4.3 GB RAM
22/04/22 22:45:27 INFO storage.BlockManagerMaster: Registered BlockManager
22/04/22 22:45:27 INFO spark.HttpServer: Starting HTTP Server
22/04/22 22:45:27 INFO server.Server: jetty-8.1.14.v20131031
22/04/22 22:45:27 INFO server.AbstractConnector: Started SocketConnector@0.0.0.0:51642
22/04/22 22:45:27 INFO broadcast.HttpBroadcast: Broadcast server started at http://192.168.1.180:51642
22/04/22 22:45:27 INFO spark.HttpFileServer: HTTP File server directory is /var/folders/ts/q9gpv9j931jgspdt_hzq6w940000gn/T/spark-b4ecedaa-464a-4fec-9d49-3214c4b33ab1
22/04/22 22:45:27 INFO spark.HttpServer: Starting HTTP Server
22/04/22 22:45:27 INFO server.Server: jetty-8.1.14.v20131031
22/04/22 22:45:27 INFO server.AbstractConnector: Started SocketConnector@0.0.0.0:51643
22/04/22 22:45:27 INFO server.Server: jetty-8.1.14.v20131031
22/04/22 22:45:27 INFO server.AbstractConnector: Started SelectChannelConnector@0.0.0.0:4040
22/04/22 22:45:27 INFO ui.SparkUI: Started SparkUI at http://192.168.1.180:4040
22/04/22 22:45:27 INFO storage.MemoryStore: ensureFreeSpace(32816) called with curMem=0, maxMem=4581438259
22/04/22 22:45:27 INFO storage.MemoryStore: Block broadcast_0 stored as values to memory (estimated size 32.0 KB, free 4.3 GB)
22/04/22 22:45:28 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
22/04/22 22:45:28 WARN snappy.LoadSnappy: Snappy native library not loaded
22/04/22 22:45:28 INFO mapred.FileInputFormat: Total input paths to process : 1
22/04/22 22:45:28 INFO spark.SparkContext: Starting job: count at TempestAnalytics.java:33
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Got job 0 (count at TempestAnalytics.java:33) with 1 output partitions (allowLocal=false)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Final stage: Stage 0(count at TempestAnalytics.java:33)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Parents of final stage: List()
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Missing parents: List()
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Submitting Stage 0 (FlatMappedRDD[2] at flatMap at TempestAnalytics.java:30), which has no missing parents
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Submitting 1 missing tasks from Stage 0 (FlatMappedRDD[2] at flatMap at TempestAnalytics.java:30)
22/04/22 22:45:28 INFO scheduler.TaskSchedulerImpl: Adding task set 0.0 with 1 tasks
22/04/22 22:45:28 INFO scheduler.TaskSetManager: Starting task 0.0:0 as TID 0 on executor localhost: localhost (PROCESS_LOCAL)
22/04/22 22:45:28 INFO scheduler.TaskSetManager: Serialized task 0.0:0 as 2253 bytes in 1 ms
22/04/22 22:45:28 INFO executor.Executor: Running task ID 0
22/04/22 22:45:28 INFO storage.BlockManager: Found block broadcast_0 locally
22/04/22 22:45:28 INFO rdd.HadoopRDD: Input split: file:/Users/pengqinzhi/Desktop/Distributed-Systems/Project/5/Project5Part2/TheTempest.txt:0+99846
22/04/22 22:45:28 INFO executor.Executor: Serialized size of result for 0 is 597
22/04/22 22:45:28 INFO executor.Executor: Sending result for 0 directly to driver
22/04/22 22:45:28 INFO executor.Executor: Finished task ID 0
22/04/22 22:45:28 INFO scheduler.TaskSetManager: Finished TID 0 in 101 ms on localhost (progress: 1/1)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Completed ResultTask(0, 0)
22/04/22 22:45:28 INFO scheduler.TaskSchedulerImpl: Removed TaskSet 0.0, whose tasks have all completed, from pool
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Stage 0 (count at TempestAnalytics.java:33) finished in 0.109 s
22/04/22 22:45:28 INFO spark.SparkContext: Job finished: count at TempestAnalytics.java:33, took 0.1997205 s
22/04/22 22:45:28 INFO spark.SparkContext: Starting job: count at TempestAnalytics.java:42
Number of lines: 3466
```

Task1

Using the split method of the java String class and the flatMap method of the JavaRDD class, use the count method of the JavaRDD class to display the number of words in The Tempest.

```
// flatMap each line to words in the line
JavaRDD<String> wordsFromFile = inputFile.flatMap(content ->
Arrays.asList(content.split("[^a-zA-Z]+")));

Function<String, Boolean> filter = k -> (!k.isEmpty());

// count number of words
long countWords = wordsFromFile.filter(filter).count();
System.out.println("Number of words: " + countWords);
```

```
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Got job 1 (count at TempestAnalytics.java:42) with 1 output partitions (allowLocal=false)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Final stage: Stage 1(count at TempestAnalytics.java:42)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Parents of final stage: List()
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Missing parents: List()
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Submitting Stage 1 (FilteredRDD[4] at filter at TempestAnalytics.java:42), which has no missing parents
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Submitting 1 missing tasks from Stage 1 (FilteredRDD[4] at filter at TempestAnalytics.java:42)
22/04/22 22:45:28 INFO scheduler.TaskSchedulerImpl: Adding task set 1.0 with 1 tasks
22/04/22 22:45:28 INFO scheduler.TaskSetManager: Starting task 1.0:0 as TID 1 on executor localhost: localhost (PROCESS_LOCAL)
22/04/22 22:45:28 INFO scheduler.TaskSetManager: Serialized task 1.0:0 as 2374 bytes in 1 ms
22/04/22 22:45:28 INFO executor.Executor: Running task ID 1
22/04/22 22:45:28 INFO storage.BlockManager: Found block broadcast_0 locally
22/04/22 22:45:28 INFO rdd.HadoopRDD: Input split: file:/Users/pengqinzhi/Desktop/Distributed-Systems/Project/5/Project5Part2/TheTempest.txt:0+99846
22/04/22 22:45:28 INFO executor.Executor: Serialized size of result for 1 is 597
22/04/22 22:45:28 INFO executor.Executor: Sending result for 1 directly to driver
22/04/22 22:45:28 INFO executor.Executor: Finished task ID 1
22/04/22 22:45:28 INFO scheduler.TaskSetManager: Finished TID 1 in 71 ms on localhost (progress: 1/1)
22/04/22 22:45:28 INFO scheduler.TaskSchedulerImpl: Removed TaskSet 1.0, whose tasks have all completed, from pool
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Completed ResultTask(1, 0)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Stage 1 (count at TempestAnalytics.java:42) finished in 0.073 s
22/04/22 22:45:28 INFO spark.SparkContext: Job finished: count at TempestAnalytics.java:42, took 0.082529333 s
Number of words: 18167
```

Task2
Using some of the work you did above and the JavaRDD distinct() and count() methods, display the number of distinct words in The Tempest.

```
// count number of distinct words
long countDistinctWords = wordsFromFile.distinct().filter(filter).count();
System.out.println("Number of distinct words: " + countDistinctWords);
```

```
22/04/22 22:45:28 INFO spark.SparkContext: Starting job: count at TempestAnalytics.java:46
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Registering RDD 6 (distinct at TempestAnalytics.java:46)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Got job 2 (count at TempestAnalytics.java:46) with 1 output partitions (allowLocal=false)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Final stage: Stage 2(count at TempestAnalytics.java:46)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Parents of final stage: List(Stage 3)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Missing parents: List(Stage 3)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Submitting Stage 3 (MapPartitionsRDD[6] at distinct at TempestAnalytics.java:46), which has no missing parents
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Submitting 1 missing tasks from Stage 3 (MapPartitionsRDD[6] at distinct at TempestAnalytics.java:46)
22/04/22 22:45:28 INFO scheduler.TaskSchedulerImpl: Adding task set 3.0 with 1 tasks
22/04/22 22:45:28 INFO scheduler.TaskSetManager: Starting task 3.0:0 as TID 2 on executor localhost: localhost (PROCESS_LOCAL)
22/04/22 22:45:28 INFO scheduler.TaskSetManager: Serialized task 3.0:0 as 2555 bytes in 0 ms
22/04/22 22:45:28 INFO executor.Executor: Running task ID 2
22/04/22 22:45:28 INFO storage.BlockManager: Found block broadcast_0 locally
22/04/22 22:45:28 INFO rdd.HadoopRDD: Input split: file:/Users/pengqinzhi/Desktop/Distributed-Systems/Project/5/Project5Part2/TheTempest.txt:0+99846
22/04/22 22:45:28 INFO executor.Executor: Serialized size of result for 2 is 783
22/04/22 22:45:28 INFO executor.Executor: Sending result for 2 directly to driver
22/04/22 22:45:28 INFO executor.Executor: Finished task ID 2
22/04/22 22:45:28 INFO scheduler.TaskSetManager: Finished TID 2 in 246 ms on localhost (progress: 1/1)
22/04/22 22:45:28 INFO scheduler.TaskSchedulerImpl: Removed TaskSet 3.0, whose tasks have all completed, from pool
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Completed ShuffleMapTask(3, 0)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Stage 3 (distinct at TempestAnalytics.java:46) finished in 0.250 s
22/04/22 22:45:28 INFO scheduler.DAGScheduler: looking for newly runnable stages
22/04/22 22:45:28 INFO scheduler.DAGScheduler: running: Set()
22/04/22 22:45:28 INFO scheduler.DAGScheduler: waiting: Set(Stage 2)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: failed: Set()
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Missing parents for Stage 2: List()
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Submitting Stage 2 (FilteredRDD[10] at filter at TempestAnalytics.java:46), which is now runnable
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Submitting 1 missing tasks from Stage 2 (FilteredRDD[10] at filter at TempestAnalytics.java:46)
22/04/22 22:45:28 INFO scheduler.TaskSchedulerImpl: Adding task set 2.0 with 1 tasks
22/04/22 22:45:28 INFO scheduler.TaskSetManager: Starting task 2.0:0 as TID 3 on executor localhost: localhost (PROCESS_LOCAL)
22/04/22 22:45:28 INFO scheduler.TaskSetManager: Serialized task 2.0:0 as 2513 bytes in 2 ms
22/04/22 22:45:28 INFO executor.Executor: Running task ID 3
22/04/22 22:45:28 INFO storage.BlockManager: Found block broadcast_0 locally
22/04/22 22:45:28 INFO storage.BlockFetcherIterator$BasicBlockFetcherIterator: maxBytesInFlight: 50331648, targetRequestSize: 10066329
22/04/22 22:45:28 INFO storage.BlockFetcherIterator$BasicBlockFetcherIterator: Getting 1 non-empty blocks out of 1 blocks
22/04/22 22:45:28 INFO storage.BlockFetcherIterator$BasicBlockFetcherIterator: Started 0 remote fetches in 2 ms
22/04/22 22:45:28 INFO executor.Executor: Serialized size of result for 3 is 863
22/04/22 22:45:28 INFO executor.Executor: Sending result for 3 directly to driver
22/04/22 22:45:28 INFO executor.Executor: Finished task ID 3
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Completed ResultTask(2, 0)
22/04/22 22:45:28 INFO scheduler.TaskSetManager: Finished TID 3 in 174 ms on localhost (progress: 1/1)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Stage 2 (count at TempestAnalytics.java:46) finished in 0.176 s
22/04/22 22:45:28 INFO scheduler.TaskSchedulerImpl: Removed TaskSet 2.0, whose tasks have all completed, from pool
22/04/22 22:45:28 INFO spark.SparkContext: Job finished: count at TempestAnalytics.java:46, took 0.48133425 s
Number of distinct words: 3595
```

Task3
Use the split method with a regular expression of "" and a flatmap to find the number of symbols in The Tempest.

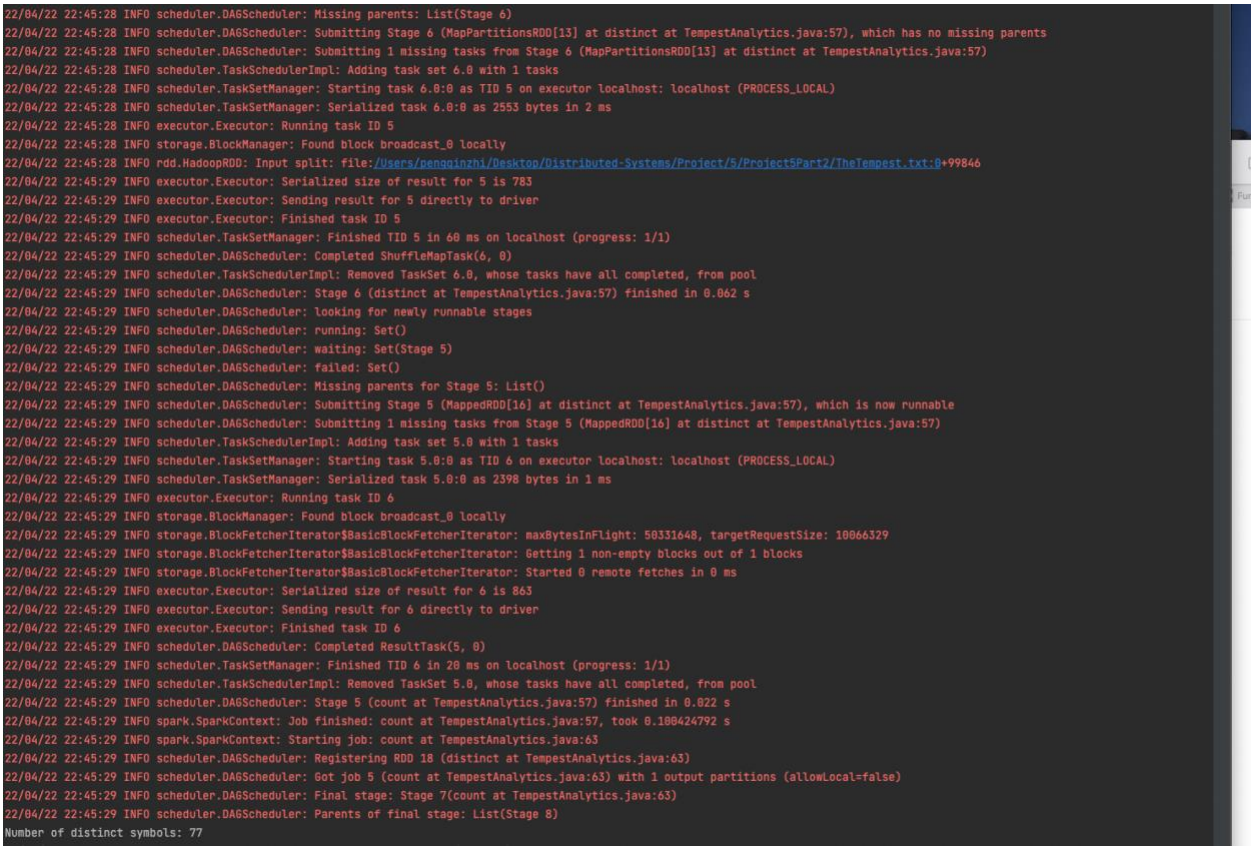
```
// flatMap each line to symbols in the line
JavaRDD<String> symbolsFromFile = inputFile.flatMap(content ->
Arrays.asList(content.split("")));

// count number of symbols
long countSymbols = symbolsFromFile.count();
System.out.println("Number of symbols: " + countSymbols);
```

```
22/04/22 22:45:28 INFO spark.SparkContext: Starting job: count at TempestAnalytics.java:53
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Got job 3 (count at TempestAnalytics.java:53) with 1 output partitions (allowLocal=false)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Final stage: Stage 4(count at TempestAnalytics.java:53)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Parents of final stage: List()
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Missing parents: List()
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Submitting Stage 4 (FlatMappedRDD[11] at flatMap at TempestAnalytics.java:50), which has no missing parents
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Submitting 1 missing tasks from Stage 4 (FlatMappedRDD[11] at flatMap at TempestAnalytics.java:50)
22/04/22 22:45:28 INFO scheduler.TaskSchedulerImpl: Adding task set 4.0 with 1 tasks
22/04/22 22:45:28 INFO scheduler.TaskSetManager: Starting task 4.0:0 as TID 4 on executor localhost: localhost (PROCESS_LOCAL)
22/04/22 22:45:28 INFO scheduler.TaskSetManager: Serialized task 4.0:0 as 2254 bytes in 2 ms
22/04/22 22:45:28 INFO executor.Executor: Running task ID 4
22/04/22 22:45:28 INFO storage.BlockManager: Found block broadcast_0 locally
22/04/22 22:45:28 INFO rdd.HadoopRDD: Input split: file:/Users/pengqinzhi/Desktop/Distributed-Systems/Project/5/Project5Part2/TheTempest.txt:0+99846
22/04/22 22:45:28 INFO executor.Executor: Serialized size of result for 4 is 597
22/04/22 22:45:28 INFO executor.Executor: Sending result for 4 directly to driver
22/04/22 22:45:28 INFO executor.Executor: Finished task ID 4
22/04/22 22:45:28 INFO scheduler.TaskSetManager: Finished TID 4 in 60 ms on localhost (progress: 1/1)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Completed ResultTask(4, 0)
22/04/22 22:45:28 INFO scheduler.TaskSchedulerImpl: Removed TaskSet 4.0, whose tasks have all completed, from pool
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Stage 4 (count at TempestAnalytics.java:53) finished in 0.062 s
22/04/22 22:45:28 INFO spark.SparkContext: Job finished: count at TempestAnalytics.java:53, took 0.06972125 s
22/04/22 22:45:28 INFO spark.SparkContext: Starting job: count at TempestAnalytics.java:57
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Registering RDD 13 (distinct at TempestAnalytics.java:57)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Got job 4 (count at TempestAnalytics.java:57) with 1 output partitions (allowLocal=false)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Final stage: Stage 5(count at TempestAnalytics.java:57)
22/04/22 22:45:28 INFO scheduler.DAGScheduler: Parents of final stage: List(Stage 6)
Number of symbols: 97132
```

Task4
Find the number of distinct symbols in The Tempest.

```
// count number of distinct symbols
long countDistinctSymbols = symbolsFromFile.distinct().count();
System.out.println("Number of distinct symbols: " + countDistinctSymbols);
```



Task5
Find the number of distinct letters in The Tempest.

```java
// count number of distinct symbols
long countDistinctSymbols = symbolsFromFile.distinct().count();
System.out.println("Number of distinct symbols: " + countDistinctSymbols);

// filter the symbols which are not letter
Function<String, Boolean> filter2 = k -> (k.toLowerCase().matches("[a-z]"));

// count number of distinct letters
long countDistinctLetters =
symbolsFromFile.distinct().filter(filter2).count();
System.out.println("Number of distinct letters: " + countDistinctLetters);
```

```
22/04/22 22:45:29 INFO scheduler.DAGScheduler: Missing parents: List(Stage 8)
22/04/22 22:45:29 INFO scheduler.DAGScheduler: Submitting Stage 8 (MapPartitionsRDD[18] at distinct at TempestAnalytics.java:63), which has no missing parents
22/04/22 22:45:29 INFO scheduler.DAGScheduler: Submitting 1 missing tasks from Stage 8 (MapPartitionsRDD[18] at distinct at TempestAnalytics.java:63)
22/04/22 22:45:29 INFO scheduler.TaskSchedulerImpl: Adding task set 8.0 with 1 tasks
22/04/22 22:45:29 INFO scheduler.TaskSetManager: Starting task 8.0:0 as TID 7 on executor localhost: localhost (PROCESS_LOCAL)
22/04/22 22:45:29 INFO scheduler.TaskSetManager: Serialized task 8.0:0 as 2555 bytes in 0 ms
22/04/22 22:45:29 INFO executor.Executor: Running task ID 7
22/04/22 22:45:29 INFO storage.BlockManager: Found block broadcast_0 locally
22/04/22 22:45:29 INFO rdd.HadoopRDD: Input split: file:/Users/pengqinzhi/Desktop/Distributed-Systems/Project/5/Project5Part2/TheTempest.txt:0+99846
22/04/22 22:45:29 INFO executor.Executor: Serialized size of result for 7 is 783
22/04/22 22:45:29 INFO executor.Executor: Sending result for 7 directly to driver
22/04/22 22:45:29 INFO executor.Executor: Finished task ID 7
22/04/22 22:45:29 INFO scheduler.DAGScheduler: Completed ShuffleMapTask(8, 0)
22/04/22 22:45:29 INFO scheduler.TaskSetManager: Finished TID 7 in 47 ms on localhost (progress: 1/1)
22/04/22 22:45:29 INFO scheduler.TaskSchedulerImpl: Removed TaskSet 8.0, whose tasks have all completed, from pool
22/04/22 22:45:29 INFO scheduler.DAGScheduler: Stage 8 (distinct at TempestAnalytics.java:63) finished in 0.048 s
22/04/22 22:45:29 INFO scheduler.DAGScheduler: looking for newly runnable stages
22/04/22 22:45:29 INFO scheduler.DAGScheduler: running: Set()
22/04/22 22:45:29 INFO scheduler.DAGScheduler: waiting: Set(Stage 7)
22/04/22 22:45:29 INFO scheduler.DAGScheduler: failed: Set()
22/04/22 22:45:29 INFO scheduler.DAGScheduler: Missing parents for Stage 7: List()
22/04/22 22:45:29 INFO scheduler.DAGScheduler: Submitting Stage 7 (FilteredRDD[22] at filter at TempestAnalytics.java:63), which is now runnable
22/04/22 22:45:29 INFO scheduler.DAGScheduler: Submitting 1 missing tasks from Stage 7 (FilteredRDD[22] at filter at TempestAnalytics.java:63)
22/04/22 22:45:29 INFO scheduler.TaskSchedulerImpl: Adding task set 7.0 with 1 tasks
22/04/22 22:45:29 INFO scheduler.TaskSetManager: Starting task 7.0:0 as TID 8 on executor localhost: localhost (PROCESS_LOCAL)
22/04/22 22:45:29 INFO scheduler.TaskSetManager: Serialized task 7.0:0 as 2513 bytes in 0 ms
22/04/22 22:45:29 INFO executor.Executor: Running task ID 8
22/04/22 22:45:29 INFO storage.BlockManager: Found block broadcast_0 locally
22/04/22 22:45:29 INFO storage.BlockFetcherIterator$BasicBlockFetcherIterator: maxBytesInFlight: 50331648, targetRequestSize: 10066329
22/04/22 22:45:29 INFO storage.BlockFetcherIterator$BasicBlockFetcherIterator: Getting 1 non-empty blocks out of 1 blocks
22/04/22 22:45:29 INFO storage.BlockFetcherIterator$BasicBlockFetcherIterator: Started 0 remote fetches in 0 ms
22/04/22 22:45:29 INFO executor.Executor: Serialized size of result for 8 is 863
22/04/22 22:45:29 INFO executor.Executor: Sending result for 8 directly to driver
22/04/22 22:45:29 INFO executor.Executor: Finished task ID 8
22/04/22 22:45:29 INFO scheduler.DAGScheduler: Completed ResultTask(7, 0)
22/04/22 22:45:29 INFO scheduler.TaskSetManager: Finished TID 8 in 13 ms on localhost (progress: 1/1)
22/04/22 22:45:29 INFO scheduler.DAGScheduler: Stage 7 (count at TempestAnalytics.java:63) finished in 0.014 s
22/04/22 22:45:29 INFO scheduler.TaskSchedulerImpl: Removed TaskSet 7.0, whose tasks have all completed, from pool
22/04/22 22:45:29 INFO spark.SparkContext: Job finished: count at TempestAnalytics.java:63, took 0.084607167 s
Number of distinct letters: 51
```

Task6
Ask your user to enter a word and show all of the lines of The Tempest that contain that word. The search will be case-sensitive.

```
// ask the user to enter a word
System.out.println("Please input a word: ");
Scanner token = new Scanner(System.in);
String input = token.nextLine();

// filter the lines which do not contain the word
Function<String, Boolean> filter3 = k -> (k.contains(input));

// show all of the lines of The Tempest that contain that word
JavaRDD<String> searchFromFile = linesFromFile.filter(filter3);

for (String line : searchFromFile.collect()) {
    System.out.println(line);
}
```

```
Please input a word:
love
22/04/22 22:45:33 INFO spark.SparkContext: Starting job: collect at TempestAnalytics.java:76
22/04/22 22:45:33 INFO scheduler.DAGScheduler: Got job 6 (collect at TempestAnalytics.java:76) with 1 output partitions (allowLocal=false)
22/04/22 22:45:33 INFO scheduler.DAGScheduler: Final stage: Stage 9(collect at TempestAnalytics.java:76)
22/04/22 22:45:33 INFO scheduler.DAGScheduler: Parents of final stage: List()
22/04/22 22:45:33 INFO scheduler.DAGScheduler: Missing parents: List()
22/04/22 22:45:33 INFO scheduler.DAGScheduler: Submitting Stage 9 (FilteredRDD[23] at filter at TempestAnalytics.java:74), which has no missing parents
22/04/22 22:45:33 INFO scheduler.DAGScheduler: Submitting 1 missing tasks from Stage 9 (FilteredRDD[23] at filter at TempestAnalytics.java:74)
22/04/22 22:45:33 INFO scheduler.TaskSchedulerImpl: Adding task set 9.0 with 1 tasks
22/04/22 22:45:33 INFO scheduler.TaskSetManager: Starting task 9.0:0 as TID 9 on executor localhost: localhost (PROCESS_LOCAL)
22/04/22 22:45:33 INFO scheduler.TaskSetManager: Serialized task 9.0:0 as 2398 bytes in 2 ms
22/04/22 22:45:33 INFO executor.Executor: Running task ID 9
22/04/22 22:45:33 INFO storage.BlockManager: Found block broadcast_0 locally
22/04/22 22:45:33 INFO rdd.HadoopRDD: Input split: file:/Users/pengqinzhi/Desktop/Distributed-Systems/Project/5/Project5Part2/TheTempest.txt:0+99846
22/04/22 22:45:33 INFO executor.Executor: Serialized size of result for 9 is 1460
22/04/22 22:45:33 INFO executor.Executor: Sending result for 9 directly to driver
22/04/22 22:45:33 INFO executor.Executor: Finished task ID 9
22/04/22 22:45:33 INFO scheduler.TaskSetManager: Finished TID 9 in 22 ms on localhost (progress: 1/1)
22/04/22 22:45:33 INFO scheduler.DAGScheduler: Completed ResultTask(9, 0)
22/04/22 22:45:33 INFO scheduler.TaskSchedulerImpl: Removed TaskSet 9.0, whose tasks have all completed, from pool
22/04/22 22:45:33 INFO scheduler.DAGScheduler: Stage 9 (collect at TempestAnalytics.java:76) finished in 0.023 s
22/04/22 22:45:33 INFO spark.SparkContext: Job finished: collect at TempestAnalytics.java:76, took 0.039568375 s
BOATSWAIN  None that I more love than myself. You are
Of all the world I loved, and to him put
So dear the love my people bore me, nor set
Knowing I loved my books, he furnished me
Into a cloven pine, within which rift
I do not love to look on.
That burn by day and night. And then I loved thee,
And I the King shall love thee.
All wound with adders, who with cloven tongues
    She loved not the savor of tar nor of pitch,
MIRANDA  Do you love me?
Do love, prize, honor you.
And his and mine loved darling. [He exits, above.]
Were but my trials of thy love, and thou
With such love as 'tis now, the murkiest den,
    Do you love me, master? No?
Whose shadow the dismissed bachelor loves,
A contract of true love to celebrate,
On the blest lovers.
A contract of true love. Be not too late.
FERDINAND  No, my dearest love,
Of these our dear-beloved solemnized,

Process finished with exit code 0
```