Lab 2 Documentation

- In this use case a text file with lots of formatted and non-formatted words and sentences is taken as shown below

SparkEvenNumberGenerator.scala × | input.txt ×

```
Data science employs techniques and theories drawn from                many fields within the broad areas of mathematics,
statistics, operations research information science, and computer science, including signal processing, probability

models, machine learning, statistical learning, data mining, database, data engineering, pattern recognition and learning, visualization,
predictive analytics, uncertainty modeling, data warehousing, data compression, computer programming, artificial intelligence, and high

performance computing. Methods that scale to big data are of particular interest in data science, although the discipline is not generally considered to be restricted to such big data, a

Data science affects academic and applied



research in many domains,                                                  including machine translation, speech recognition,
robotics, search engines, digital economy, but also                      the biological sciences, medical informatics, health care, social sciences
and the humanities. It heavily influences economics, ,



business and                                                finance.
From the business perspective, data science is an integral part of competitive intelligence,
a newly emerging field that encompasses a number of activities,                              such as data mining and data analysis
```

- The spark program takes this input and displays the words that are of even length. So it basically filters out all the words that are of odd length.
- The execution of the program can be seen below.

```
"C:\Program Files\Java\jdk1.8.0_101\bin\java" ...
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
16/09/07 21:04:29 INFO SparkContext: Running Spark version 1.4.1
16/09/07 21:04:30 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
16/09/07 21:04:30 INFO SecurityManager: Changing view acls to: prane
16/09/07 21:04:30 INFO SecurityManager: Changing modify acls to: prane
16/09/07 21:04:30 INFO SecurityManager: SecurityManager: authentication disabled; ui acls disabled; users with view permissions: Set(prane); users with
16/09/07 21:04:31 INFO Slf4jLogger: Slf4jLogger started
16/09/07 21:04:31 INFO Remoting: Starting remoting
16/09/07 21:04:31 INFO Remoting: Remoting started; listening on addresses :[akka.tcp://sparkDriver@192.168.1.191:52068]
16/09/07 21:04:31 INFO Utils: Successfully started service 'sparkDriver' on port 52068.
16/09/07 21:04:31 INFO SparkEnv: Registering MapOutputTracker
16/09/07 21:04:31 INFO SparkEnv: Registering BlockManagerMaster
16/09/07 21:04:31 INFO DiskBlockManager: Created local directory at C:\Users\prane\AppData\Local\Temp\spark-e47e9fbc-df4e-4d32-92de-95fe69cc94fe\blockmg
16/09/07 21:04:31 INFO MemoryStore: MemoryStore started with capacity 973.6 MB
16/09/07 21:04:31 INFO HttpFileServer: HTTP File server directory is C:\Users\prane\AppData\Local\Temp\spark-e47e9fbc-df4e-4d32-92de-95fe69cc94fe\httpd-
16/09/07 21:04:31 INFO HttpServer: Starting HTTP Server
16/09/07 21:04:32 INFO Utils: Successfully started service 'HTTP file server' on port 52069.
16/09/07 21:04:32 INFO SparkEnv: Registering OutputCommitCoordinator
16/09/07 21:04:32 INFO Utils: Successfully started service 'SparkUI' on port 4040.
16/09/07 21:04:32 INFO SparkUI: Started SparkUI at http://192.168.1.191:4040
16/09/07 21:04:32 INFO Executor: Starting executor ID driver on host localhost
16/09/07 21:04:32 INFO Utils: Successfully started service 'org.apache.spark.network.netty.NettyBlockTransferService' on port 52089.
16/09/07 21:04:32 INFO NettyBlockTransferService: Server created on 52089
16/09/07 21:04:32 INFO BlockManagerMaster: Trying to register BlockManager
16/09/07 21:04:32 INFO BlockManagerMasterEndpoint: Registering block manager localhost:52089 with 973.6 MB RAM, BlockManagerId(driver, localhost, 52089)
16/09/07 21:04:32 INFO BlockManagerMaster: Registered BlockManager
16/09/07 21:04:34 INFO MemoryStore: ensureFreeSpace(110248) called with curMem=0, maxMem=1020914565
16/09/07 21:04:34 INFO MemoryStore: Block broadcast_0 stored as values in memory (estimated size 107.7 KB, free 973.5 MB)
16/09/07 21:04:34 INFO MemoryStore: ensureFreeSpace(10090) called with curMem=110248, maxMem=1020914565
16/09/07 21:04:34 INFO MemoryStore: Block broadcast_0_piece0 stored as bytes in memory (estimated size 9.9 KB, free 973.5 MB)
```

Platform and
IntelliJ IDEA is

- The output is displayed on the console and also stored in the text files.
- The console output can be found below.

```
Data
techniques
theories
from
many
fields
within
of
statistics
operations
research
computer
signal
processing
models
learning
learning
data
mining
database
data
learning
predictive
modeling
data
data
computer
artificial
intelligence
high
computing
that
to
data
of
particular
```

- The part files output can be found below.
- Two part files are generated by Spark for our input.

- Part-file0

SparkEvenNumberGenerator.scala × | input.txt × | part-00000 ×

```
Data
techniques
theories
from
many
fields
within
of
statistics
operations
research
computer
signal
processing
models
learning
learning
data
mining
database
data
learning
predictive
modeling
data
data
computer
artificial
intelligence
high
computing
that
to
data
of
particular
```

- Part-file1

SparkEvenNumberGenerator.scala × | input.txt × | part-00000 × | part-00001

```
Data
academic
research
in
many
speech
robotics
search
also
biological
sciences
health
care
social
sciences
It
influences
business
finance
From
business
data
is
an
integral
part
of
intelligence
emerging
that
number
of
activities
such
as
data
```

- As required by the assignment I have used at least 2 transformations and 2 actions.
- Transformations used are:
    1. **map:**
       val stripped_data = main_filter.map(e =>
       e.stripPrefix(",").stripSuffix(",").stripPrefix(".").stripSuffix(".").trim)
       this strips of the comma or period appended/prepended to the output

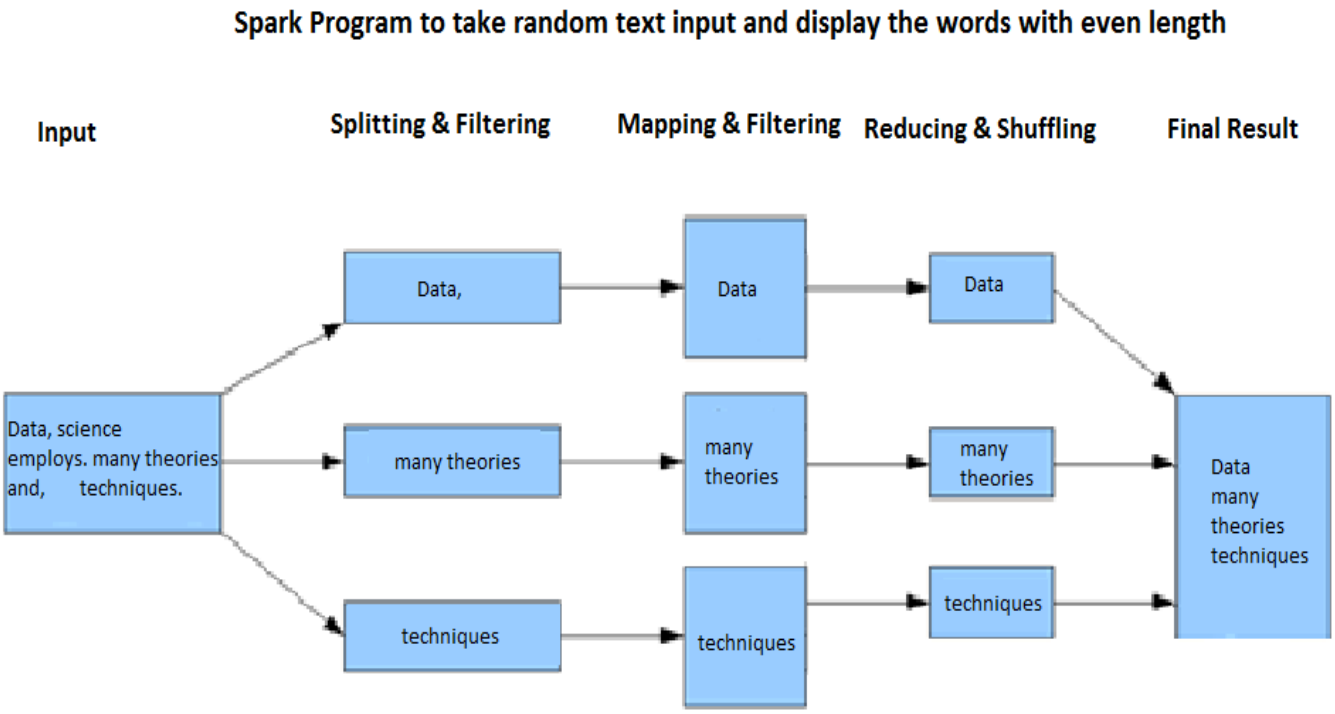
    2. **flatMap:**
       val fil_input = input.flatMap((line=>{line.split(" ").filter(_.nonEmpty)}))
       this takes the input file and goes line by line and splits the tokens using " "(space) as a delimiter. It also filters out the empty tokens

    3. **filter:**
       val main_filter = (fil_input.filter(e => e.stripPrefix(",").stripSuffix(",").trim.length%2==0))
       In this step the filtering takes place such that only the words with even length are stored in our main_filter

- Actions are:
    1. **collect:**
       val console_data = stripped_data.collect()
       in order to view the content of the RDD, in other words to output on the console we use collect()

    2. **saveAsTextFile:**
       stripped_data.saveAsTextFile("output")
       this saves the output in the output folder in part files.

- **Map-Reduce Diagram**



- **Brief Explanation of the above diagram:**
    1. The input is a random text file containing formatted and non-formatted text (refer the diagram on page 1). The input has the words appended with comma or period or there are multiple spaces between the words. The sentences are not consistent. So this is truly a random text data.
    2. Our program goes line by line and extracts the words from the line. In addition to that it filters out the empty strings and also the words that are of odd length. So now we have the even length words but they have extra attributes like there might be a comma or period or spaces appended or prepended to them.
    3. In the mapping stage we map each word to its word minus the extra attributes mentioned above.
    4. The reduce and the shuffling phase doesn't do much in our program but it is essential part of map reduce paradigm.
    5. Finally, the output from the reducers is combined and we get our final output data that is the words with even length.