Session 24:
SPARK STREAMING
Assignment 1

Task 1

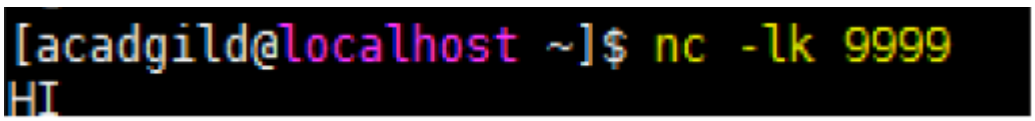
Read a stream of Strings, fetch the words which can be converted to numbers. Filter out the rows, where the sum of numbers in that line is odd.
Provide the sum of all the remaining numbers in that batch.

Command,

sudo yum install nc

```
[acadgild@localhost ~]$ sudo yum install nc
Loaded plugins: fastestmirror, refresh-packagekit, security
Setting up Install Process
Loading mirror speeds from cached hostfile
 * base: ftp.iitm.ac.in
 * extras: ftp.iitm.ac.in
 * updates: ftp.iitm.ac.in
Resolving Dependencies
--> Running transaction check
---> Package nc.i686 0:1.84-24.el6 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package                Arch                Version                 Repository                Size
================================================================================
Installing:
 nc                     i686                1.84-24.el6             base                      57 k

Transaction Summary
================================================================================
Install       1 Package(s)

Total download size: 57 k
Installed size: 107 k
Is this ok [y/N]: y
Downloading Packages:
nc-1.84-24.el6.i686.rpm                                  |  57 kB     00:00
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing : nc-1.84-24.el6.i686                                         1/1
  Verifying  : nc-1.84-24.el6.i686                                         1/1

Installed:
  nc.i686 0:1.84-24.el6

Complete!
[acadgild@localhost ~]$ nc -lk 9999
```

nc –lk 9999



/home/acadgild/spark-2.2.1-bin-hadoop2.7/bin/spark-shell --master local[4]

import org.apache.spark._

import org.apache.spark.streaming._

import org.apache.spark.streaming.StreamingContext._

```
scala> import org.apache.spark._
import org.apache.spark._

scala> import org.apache.spark.streaming._
import org.apache.spark.streaming._

scala> import org.apache.spark.streaming.StreamingContext._
import org.apache.spark.streaming.StreamingContext._
```

val EvenLines = sc.accumulator(0)

```
scala> val EvenLines = sc.accumulator(0)
warning: there were two deprecation warnings; re-run with -deprecation for details
EvenLines: org.apache.spark.Accumulator[Int] = 0
```

val wordstonumbers = map("Hi"->1, "This"->2, "is"->3, "Assignment"->4, "number"->5, "Twenty"->6,"it"->7, "about"->8, "spark"->9, "Streaming"->10)

val wordstonumbersbroadcast = sc.broadcast(wordstonumbers)

```
scala> val wordstonumbers = Map("Hi" -> 1, "This" -> 2, "is" ->3, "Assignment"->4, "number"->5, "Twenty"->6, "it"->7, "about"->8, "spark"->9, "
Streaming"->10)
wordstonumbers: scala.collection.immutable.Map[String,Int] = Map(number -> 5, is -> 3, This -> 2, Streaming -> 10, it -> 7, Twenty -> 6, spark
-> 9, Hi -> 1, Assignment -> 4, about -> 8)

scala> val wordstonumbersbroadcast = sc.broadcast(wordstonumbers)
18/01/15 12:54:49 WARN SizeEstimator: Failed to check whether UseCompressedOops is set; assuming yes
wordstonumbersbroadcast: org.apache.spark.broadcast.Broadcast[scala.collection.immutable.Map[String,Int]] = Broadcast(0)
```

def lineWordNumberSum(line:String):Int = {
var sum:Int = 0
var words = line.split(" ")
for (word <- words) sum += wordstonumbersbroadcast.value.get(word).getOrElse(0)
sum
}

```
scala> def lineWordNumberSum(line:String):Int = {
     |     var sum:Int = 0
     |     var words = line.split(" ")
     |     for (word <- words) sum += wordstonumbersbroadcast.value.get(word).getOrElse(0)
     |     sum
     | }
lineWordNumberSum: (line: String)Int
```

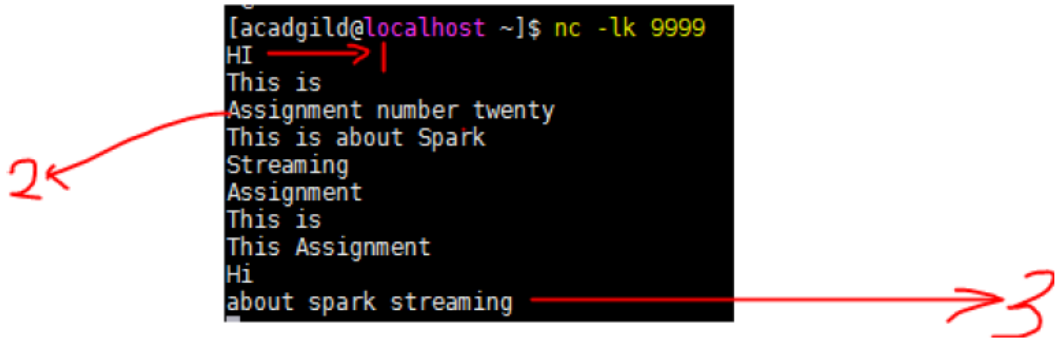
val ssc = new StreamingContext(sc, Seconds(5))
val stream = ssc.socketTextStream("localhost", 9999)

```
scala> val ssc = new StreamingContext(sc, Seconds(5))
ssc: org.apache.spark.streaming.StreamingContext = org.apache.spark.streaming.StreamingContext@af1ce8

scala> val stream = ssc.socketTextStream("localhost", 9999)
stream: org.apache.spark.streaming.dstream.ReceiverInputDStream[String] = org.apache.spark.streaming.dstream.SocketInputDStream@93161a
```

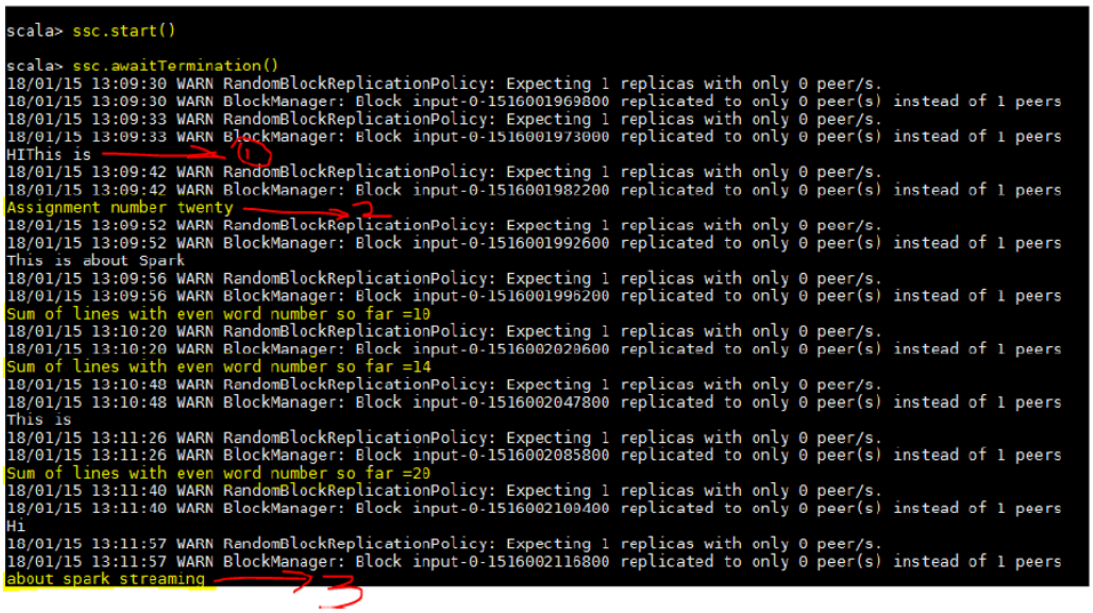
stream.foreachRDD(line => {val lineStr = line.collect().toList.mkString("")
if (lineStr != "") {var numTotal = lineWordNumberSum(lineStr) if (numTotal % 2 == 1)
println(lineStr)
else
{EvenLines += numTotal
println("Sum of lines with even word number so far =" + EvenLines.value.toInt)}}
})

```
scala> stream.foreachRDD(line => {
     |       val lineStr = line.collect().toList.mkString("")
     |       if (lineStr != "") {
     |           var numTotal = lineWordNumberSum(lineStr)
     |           if (numTotal % 2 == 1) println(lineStr)
     |           else {
     |               EvenLines += numTotal
     |               println("Sum of lines with even word number so far =" + EvenLines.value.toInt)
     |           }
     |       }
     | } )
```

ssc.start()
ssc.awaitTermination()



Output



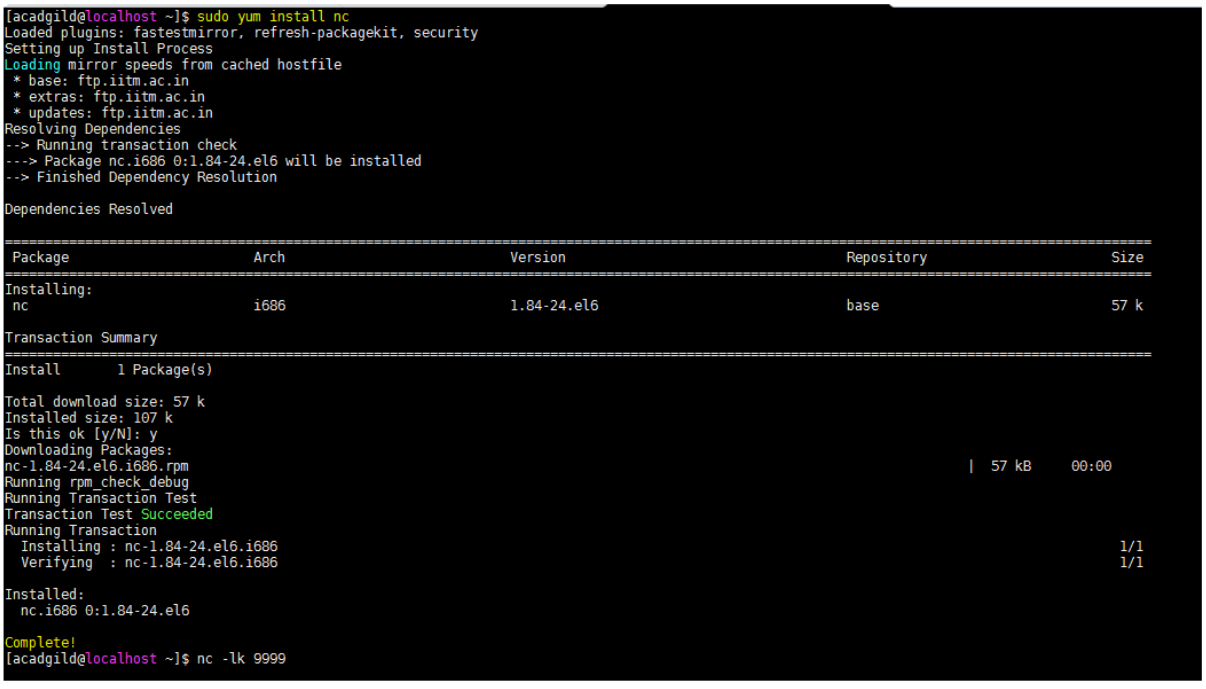
Task 2

Read two streams

1. List of strings input by user
2. Real-time set of offensive words

Find the word count of the offensive words inputted by the user as per the real-time set of offensive words

sudo yum install nc



nc –lk 9999

```
^C
[acadgild@localhost ~]$ nc -lk 9999
HI
```

/home/acadgild/spark-2.2.1-bin-hadoop2.7/bin/spark-shell --master local[4]

import org.apache.spark._

import org.apache.spark.streaming._

import org.apache.spark.streaming.StreamingContext._

```
scala> import org.apache.spark._
import org.apache.spark._

scala> import org.apache.spark.streaming._
import org.apache.spark.streaming._

scala> import org.apache.spark.streaming.StreamingContext._
import org.apache.spark.streaming.StreamingContext._
```