**CIND 719 – Assignment 3**

*Brendan Dagys*

**QUESTION 1**

nums = sc.textFile('/user/lab/number\_list.txt')

COUNT OF EVEN NUMBERS

nums.map(lambda x: int(x) % 2 == 0).sum()



COUNT OF ODD NUMBERS

nums.map(lambda x: int(x) % 2 != 0).sum()



**QUESTION 2**

**I first convert all characters to lower case. Then, I introduce multiple replace statements to get rid of special characters that could affect the result of the query, especially when generating the ‘lowest’ word counts.**

words = sc.textFile('/user/lab/shakespeare.txt') \

.flatMap(lambda line: line.lower().split()) \

.map(lambda line: line.replace('-', '')) \

.map(lambda line: line.replace('.', '')) \

.map(lambda line: line.replace(',', '')) \

.map(lambda line: line.replace(':', '')) \

.map(lambda line: line.replace(';', '')) \

.map(lambda line: line.replace('?', '')) \

.map(lambda line: line.replace('!', '')) \

.map(lambda line: line.replace('\*', '')) \

.map(lambda line: line.replace('&', '')) \

.map(lambda line: line.replace(']', '')) \

.map(lambda line: line.replace('[', '')) \

.map(lambda line: line.replace('\_', '')) \

.map(lambda word: (word, 1)) \

.reduceByKey(lambda a, b: a + b)

HIGHEST

words.takeOrdered(10, lambda x: -x[1])



LOWEST

words.takeOrdered(10, lambda x: x[1])



**QUESTION 3**

tweets = sc.textFile('/user/lab/full\_text.txt') \

.map(lambda line: (line.split('\t')[0], 1)) \

.reduceByKey(lambda a, b: a + b)

FIRST 10 USERS, WITH COUNT OF THEIR TWEETS

tweets.take(10)



OUTPUT SAVED AS TEXT FILE IN ‘/user/lab/number\_of\_tweets\_per\_user’

tweets.saveAsTextFile("/user/lab/number\_of\_tweets\_per\_user")

