Spark Training questions + answers

Lior Berlin 206363236 Nika Klimenchuk 322997628

Exercise 1

Q1. Please put your code here:

```
if __name__ == "__main__":
    if len(sys.argv) != 2:
        print("Usage: wordcount <input_folder>", file=sys.stderr)
        sys.exit(-1)

    conf = SparkConf().setAppName("python-word-count")
    sc = SparkContext(conf=conf)

    text_file = sc.textFile("hdfs://" + sys.argv[1])
    counts = text_file.flatMap(lambda line: line.split(" ")) \
                      .map(lambda word: (word, 1)) \
                      .repartition(5) \
                      .reduceByKey(lambda a, b: a + b) \
                      .filter(lambda x: len(x[0]) > 5)

    list = counts.takeOrdered(40, key = lambda x: -x[1])
    print("----------------------------------------")
    print(*list, sep="\n")
    print("----------------------------------------")
```

Q2. Add print-screen of the stage proving you have 5 tasks

Stage Id	Description		Submitted	Duration	Tasks: Succeeded/Total	Input	Output	Shuffle Read	Shuffle Write
2	takeOrdered at /home/hadoop/course/spark-word-count.py:20	+details	2025/12/17 16:36:33	0.1 s	5/5			1433.1 KiB	
1	reduceByKey at /home/hadoop/course/spark-word-count.py:14	+details	2025/12/17 16:36:32	1 s	5/5			7.0 MiB	1433.1 KiB
0	coalesce at NativeMethodAccessorImpl.java:0	+details	2025/12/17 16:36:29	2 s	3/3	6.5 MiB			7.0 MiB

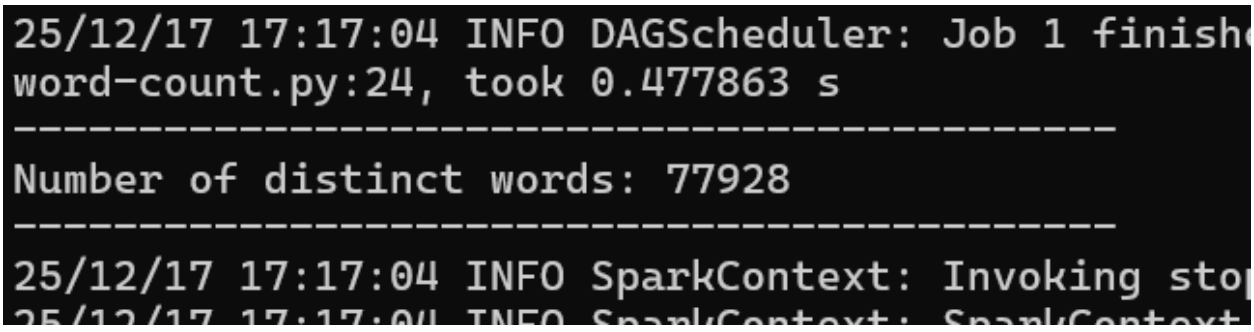
Exercise 2

Q1. Please put your code here:

```
import sys
from pyspark import SparkContext, SparkConf

if __name__ == "__main__":
    if len(sys.argv) != 2:
        print("Usage: wordcount <input_folder>", file=sys.stderr)
        sys.exit(-1)

    conf = SparkConf().setAppName("python-word-count")
    sc = SparkContext(conf=conf)

    text_file = sc.textFile("hdfs://" + sys.argv[1])
    words = text_file.flatMap(lambda line: line.split(" ")).cache()

    counts = words.map(lambda word: (word, 1)) \
                .repartition(5) \
                .reduceByKey(lambda a, b: a + b)

    list_ordered_40 = counts.takeOrdered(40, key=lambda x: -x[1])
    print("-----------------------------------------")
    print(*list_ordered_40, sep="\n")
    print("-----------------------------------------")

    distinct_words = words.distinct().count()
    print("-----------------------------------------")
    print("Number of distinct words:", distinct_words)
    print("-----------------------------------------")
```

Q2. Write the number of words found

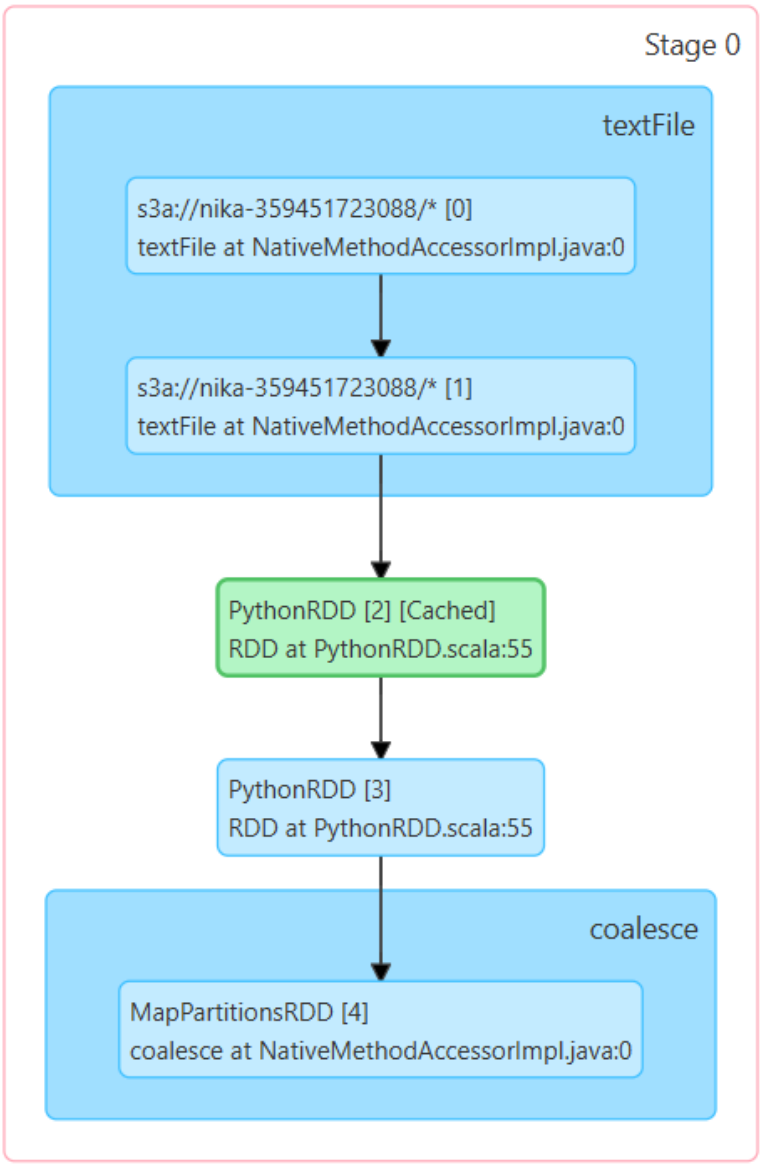
77928



Exercise 3

Put a print-scrin with the DAG of the first stage, which shows it reads the files from s3a://<your_bucket_name>

▼ DAG Visualization



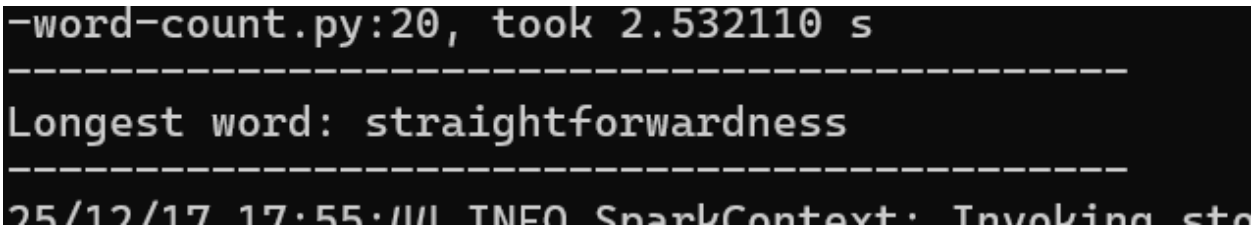
Exercise 4

Q1. Please put your code here:

```
import sys
from pyspark import SparkContext, SparkConf

if __name__ == "__main__":
    if len(sys.argv) != 2:
        print("Usage: wordcount <input_folder>", file=sys.stderr)
        sys.exit(-1)

    conf = SparkConf().setAppName("python-word-count")
    sc = SparkContext(conf=conf)

    text_file = sc.textFile("s3a://" + sys.argv[1] + "/*")

    words = text_file.flatMap(lambda line: line.split(" "))

    cleaned_words = words.map(lambda w: w.rstrip(".,")).filter(lambda w: w != "")

    alpha_words = cleaned_words.filter(lambda w: w.isalpha())

    longest_word = alpha_words.reduce(lambda a, b: a if len(a) >= len(b) else b)

    print("-------------------------------------------")
    print("Longest word:", longest_word)
    print("-------------------------------------------")
```

Q2. Put here the printout of the longest word:

Straightforwardness



Exercise 5

Q1. Please put your code here:

```
import sys
from pyspark import SparkContext, SparkConf

if __name__ == "__main__":
    if len(sys.argv) != 2:
        print("Usage: count_words_in_line <input_bucket>", file=sys.stderr)
        sys.exit(-1)

    conf = SparkConf().setAppName("count-words-in-line")
    sc = SparkContext(conf=conf)

    text_file = sc.textFile("s3a://" + sys.argv[1] + "/*")

    line_counts = text_file.map(lambda line: (len([w for w in line.split(" ") if w != ""]), line))

    max_line = line_counts.reduce(lambda a, b: a if a[0] >= b[0] else b)

    print("-----------------------------------------")
    print("Max words in a line:", max_line[0])
    print("The line:", max_line[1])
    print("-----------------------------------------")
```

Q2. Put here the printout of the line with the most words:

```
_words_in_line.py:16, took 2.332568 s
-----------------------------------------
Max words in a line: 51
The line: Archimedes, on the centre of gravity [Footnote 9: The works of Archimedes were
 not printed during Leonardo's life-time.]; anatomy [Footnote 10: Compare No. 1494.] Ale
ssandro Benedetto; The Dante of Niccolo della Croce; Inflate the lungs of a pig and obse
rve whether they increase in width and in length, or in width
-----------------------------------------
25/12/17 18:21:08 INFO SparkContext: Invoking stop() from shutdown hook
```