**MapReduce Streaming**

**Question 1: What do you need to define for processing data with Hadoop Streaming on the Map phase:**

Input records format

Correct: Yes, a mapper program should know how to parse input records

Input record processor

Correct: Yes, that is the main payload of a mapper, though Hadoop provides the default mapper implementation

Output records format

Correct: Yes, a mapper program should output records by itself

**Question 2: What you have to define for processing data with Hadoop Streaming on the Reduce phase:**

Input records format

Correct: That's right, the reducer program should parse input records

Aggregation records by key

Correct: Yes, grouping and aggregation records by key is defined in the reducer

Processor of values with the same key

Correct: Yes, that is the main payload of а reducer a user defines how to group the records by a key and how to reduce the records with equal keys

Output records format

Correct: That's right, a reducer program outputs records by itself

**Question 3: In Hadoop Streaming a mapper is run on:**

Stream of input records

Correct: Yes, a mapper reads input records one by another from stdin

**Question 4: In Hadoop Streaming a reducer is run on:**

Stream of input records

Correct: Yes, a reducer reads input records one by another from stdin

**Question 5: What phase of MapReduce is this code more suitable for?**

**!/usr/bin/env python**

**import sys**

**current\_id = None**

**value = ''**

**for line in sys.stdin:**

**new\_id, value = line.strip().split('\t', 1)**

**if new\_id != current\_id:**

**if current\_id:**

**print "%s\t%s" % (current\_id, value)**

**current\_id = new\_id**

**if current\_id:**

**print "%s\t%s" % (current\_id, value)**

Reduce

Correct: Yes, it leaves only one record from all the records with the same key (id), it's suitable for a reducer because it requires the sorted input records

**Question 6: What phase of MapReduce is this code more suitable for?**

**!/usr/bin/env python**

**import sys**

**import random**

**random.seed(100)**

**probability = float(sys.argv[1])**

**for line in sys.stdin:**

**if random.random() <= probability:**

**print line.strip()**

Map

Correct: Yes, it filters the input records (makes a sample) without a requirement that input records are sorted

**Question 7: What function is implemented in the following mapper:**

**!/usr/bin/env python**

**import sys**

**for line in sys.stdin:**

**key, value = line.strip().split('\t', 1)**

**value = int(value)**

**print "%s\t%d" % (key, value\*value)**

pow

Correct: That's right, it calculates the squares of the input values (value\*value)

**Question 8: What function is implemented in the following reducer:**

**!/usr/bin/env python**

**import sys**

**current\_key = None**

**for line in sys.stdin:**

**key = line.strip()**

**if key != current\_key:**

**if current\_key:**

**print current\_key**

**current\_key = key**

**if current\_key:**

**print current\_key**

uniq

Correct: That's right, this reducer makes the input records unique

**Question 9: How can the Reduce phase in Hadoop Streaming be omitted?**

Set the number of reducers to 0

Correct: Yes, that turns off the Reduce phase and an output of the Map phase becomes an output of the job

**Question 10: What is a Distributed Cache in Hadoop used for?**

To deliver the required files to the nodes

Correct: That’s right, that is exactly what a Distributed Cache does

**Question 11: You have the WordCount program for Hadoop, it outputs the result in the format: word count And now you want to count the total number of unique words in the text. What changes do you need to make?**

Use Hadoop counters from the existing job

Correct: Yes, counters are suitable to calculate some statistics

**Question 12: How do you pass any parameter into your Hadoop Streaming mapper script?**

Both methods are possible

Correct: Yes, you can specify arguments for your mapper script or pass them in the environment variables

**Question 13: How do you output some debug messages for you MapReduce scripts?**

Both methods are possible

Correct: Yes, it's possible to update the task status in ResourceManager Web UI and to output something in the task error log