Number: Name:

Processamento de Streams

FCT NOVA
8th May 2021 - TEST 1
closed book ; duration: 2h00

In the following questions, mark each of sentences as (**T**)rue or (**F**)alse. **Justify/Explain** separately the answers you have marked as **FALSE** in the last page of the test.

Question 1

1. ____ Map Reduce is both a programming model and companion library implementation, designed at Google, to process huge datasets in a distributed fashion, by taking advantage of specialized hardware.
2. ____ In Map Reduce, each machine participating in the *map* phase, can emit multiple tuples of different keys.
3. ____ In Map Reduce, tuples emitted in the *map* phase that share the same key, assuming there are no failures, are sent to the same machine for the *reduce* phase.
4. ____ In Map Reduce, computations are fast, because intermediate results produced in the *map* phase are kept in memory, until they are needed in the *reduce* phase.

Question 2

1. ____ Spark can perform the same kind of distributed computations that Map Reduce was designed for, only faster.
2. ____ Spark is usually much faster than Map-Reduce mainly because intermediate computations results are written to local disk storage, while Map-Reduce uses distributed storage for the same purpose.
3. ____ Spark programs represent results and intermediate computation results as RDDs (Resilient Distributed Dataset), which are immutable and each resides in a single machine.
4. ____ A fundamental aspect of Spark programs is that they encode a *dependency graph* between RDDs. The *dependency graph* is important of deciding where to place computation tasks and in the recovery from failures.

Question 3

1. ____ Spark Streaming is an evolution of Spark for processing data streams, with low latency, by allowing RDDs to grow as new data arrives.
2. ____ Spark Streaming processes streams by splitting incoming data in fixed time intervals, which can produce RDDs of variable sizes.
3. ____ Spark Streaming provides "operators"to perform arbitrary aggregations over sliding windows that operate on timestamps defined at the source of the data.
4. ____ In Spark Streaming, in computations involving sliding windows, it is an unavoidable fact that the larger the window, the longer it takes to update the result, as more data needs to be evaluated.

Question 4

1. ____ Spark SQL ditches (discards) RDDs entirely, and uses Dataframes, as the underlying core abstraction.
2. ____ Spark Dataframes organize data in tables, whose columns are named and typed, according to a *schema*.
3. ____ Using Dataframes can lead to shorter and more readable programs, at the expense of some lost computation speed.
4. ____ While Spark supports both SQL and a Dataframes based DSL[1] programming, these styles cannot be mixed in the same program.

Question 5

1. ____ Spark Structured Streaming models incoming data, as a table that is being continuously appended. Computations behave like batch-computations that are executed incrementally to take into account what has changed since the last execution.
2. ____ Spark Structured Streaming provides multiple output modes. The difference between them determines which columns of the result are sent to the output *sink*.
3. ____ Spark Structured Streaming supports sliding windows defined over arbitrary timestamps found in the original stream data.
4. ____ Spark Streaming programming model has no concept of *late events*.

[1] Domain Specific Language

Question 6

1. ____ Flume can be used to aggregate data sources prior to processing them in Spark Streaming.
2. ____ Kafka can be used to aggregate heteregenous data sources prior to processing them in Spark Streaming.
3. ____ Kafka is fault-tolerant, whereas Flume is not.
4. ____ Kafka can be used to do stream processing using a number of "native"APIs. Of these, the more expressive API is named KSQL.
5. ____ Spark Streaming provides a *continuousmode* where processing is not mini-batch based but tuple oriented, just like in Storm.
6. ____ Storm's fault-tolerance handles nodes failures via an *ack* mechanism.
7. ____ Trident is an evolution of Storm that support *exactlyonce* processing semantics.
8. ____ Flink computations are modeled as $dataflows$ that form directed cyclic graphs.
9. ____ Flink supports batch computations, as well as stream computations.
10. ____ HDFS is a fault-tolerant distributed storage solution. The HDFS client (an application) actively participates in the replication algorithm, when the file is written.
11. ____ HDFS exposes file block locations to its clients.
12. ____ Time series databases, such as InfluxDB2, often leverage LSM-tree data structures to optimize for a high level of read operations.
13. ____ ACID guarantees are one of the major design concerns of a time series database.
14. ____ A cloud-centric IoT solution leverages standard stream processing systems hosted in central data centers. One of its advantages is optimal network utilization.

Of the following **three** questions, choose **two**.

Question 7

In stream processing, it is often necessary to express computations that compare short term trends in the data to long term trends, or the stream data as a whole, for instance, a short term average vs. the average so far. Discuss what features Spark provides to this end and what strategies are employed to handle node failures and catch-up quickly.

Question 8
Spark Structured Streaming includes an explicit mechanism for handling late events. Explain what it is and discuss how late events can or cannot affect the results of a streaming computation.

Question 9
Spark and Kafka together can be used to implement stream processing with $exactly-once$ semantics. Discuss what Kafka provides that Spark requires to enable this feature.

Use this space to provide justifications for questions 1 through 6