四、简答题

1、 简述Spark SQL提供的三个功能。 （易）

参考答案:

（1）Spark SQL可以从各种结构化数据源（例如JSON、Hive、Parquet等）中读取数据，进行数据分析。 （1分）

（2）Spark SQL包含行业标准的JDBC和ODBC连接方式，因此它不局限于在Spark程序内使用SQL语句进行查询。（2分）

（3）Spark SQL可以⽆缝地将SQL查询与Spark程序进行结合，它能够将结构化数据作为Spark中的分布式数据集（RDD）进行查询，在Python、Scala和Java 中均集成了相关API，这种紧密的集成方式能够轻松地运行SQL查询以及复杂的分析算法。 （2分）

2、 简述RDD的依赖关系。 （易）

参考答案:

窄依赖是指⽗RDD的每⼀个分区最多被⼀个子RDD的分区使用，即OneToOneDependencies。窄依赖的表现⼀般分为两类，第⼀类表现为⼀个⽗RDD的分区对

应于⼀个子RDD的分区；第二类表现为多个⽗RDD的分区对应于⼀个子RDD的分区。也就是说，⼀个⽗RDD的⼀个分区不可能对应⼀个子RDD的多个分区。为了便于理解，我们通常把窄依赖形象的⽐喻为独生子⼥。当RDD执行map、filter及union和join操作时，都会产生窄依赖。 （3分）

宽依赖是指子RDD的每⼀个分区都会使用所有⽗RDD的所有分区或多个分区，即OneToManyDependecies。为了便于理解，我们通常把宽依赖形象的⽐喻为超生。当RDD做groupByKey和join操作时，会产生宽依赖。 （2分）

3、简述RDD的创建方式。 （中）

参考答案:

Spark提供了两种创建RDD的方式（1分），分别是从⽂件系统（本地和HDFS）中加载数据创建RDD（2分）和通过并行集合创建RDD。（2分）

4、 简述什么是Kafka的分区。 （中）

参考答案:

在物理意义上可以把主题看作是分区的⽇志⽂件，每个分区都是有序的，不可变的记录序列，新的消息会不断地追加到⽇志中，分区中的每条消息都会按照时间顺序分配⼀个递增的顺序编号

（每个点1分，答够5个点即可得满分）

5、 解释说明Kafka的Zookeeper是什么，是否可以在没有Zookeeper的情况下使用Kafka。 （中）

参考答案:

Zookeeper是⼀个开放源码的、高性能的协调服务，它用于Kafka的分布式应用，Zookeeper主要用于在集群中不同节点之间进行通信，（2分）使用Kafka集群时，必须部署Zookeeper服务。 （3分）

6、 简述Spark SQL的工作流程。 （中）

参考答案:

1.在解析SQL语句之前，会创建SparkSession，涉及到表名、字段名称和字段类型的元数据都将保存在SessionCatalog中；

2.当调用SparkSession的sql()方法时就会使用SparkSqlParser进行解析SQL语句，解析过程中使用的ANTLR进行词法解析和语法解析；

3.接着使用Analyzer分析器绑定逻辑计划，在该阶段，Analyzer会使用Analyzer Rules，并结合SessionCatalog，对未绑定的逻辑计划进行解析，生成已绑定的逻辑计划；

4.然后使用Optimizer优化器优化逻辑计划，该优化器同样定义了⼀套规则（Rules），利用这些规则对逻辑计划和语句进行迭代处理；

5.接着使用SparkPlanner对优化后的逻辑计划进行转换，生成可以执行的物理计划SparkPlan；

6.最终使用QueryExecution执行物理计划，此时则调用SparkPlan的execute()方法，返回RDDs。

（每个点1分，答够5个点即可得满分）

7、 简述RDD分区的作用。 （较难）

参考答案:

在分布式程序中，⽹络通信的开销是很大的，因此控制数据分布以获得最少的⽹络传输可以极大的提升整体性能，Spark程序可以通过控制RDD分区方式来减少通信开销。（2分）

Spark中所有的RDD都可以进行分区，系统会根据⼀个针对键的函数对元素进行分区。虽然Spark不能控制每个键具体划分到哪个节点上，但是可以确保相同的键出现在同⼀个分区上。 （3分）

8、简述设置检查点方式进行故障恢复。 （较难）

参考答案:

设置检查点（checkPoint）方式，本质上是将RDD写⼊磁盘进行存储。当RDD在进行宽依赖运算时，只需要在中间阶段设置⼀个检查点进行容错，即通过Spark 中的sparkContext对象调用setCheckpoint()方法，设置⼀个容错⽂件系统目录（如HDFS）作为检查点checkpoint。（2分）

将checkpoint的数据写⼊之前设置的容错⽂件系统中进行高可用的持久化存储，若是后面有节点出现宕机导致分区数据丢失，则可以从做检查点的RDD开始重新计算即可，不需要进行从头到尾的计算，这样就会减少开销。 （3分）

9、 简述利用Spark Streaming完成实时计算的工作流程。 （较难）

参考答案:

当Spark Streaming从数据源获取数据之后，则可以使用诸如map、reduce、join和window等高级函数进行复杂的计算处理（2分），最后将处理的结果存储到分布式⽂件系统、数据库中，最终利用实时Web仪表板进行展示。 （3分）

10、简述RDD转换DataFrame的两种方法。 （较难）

参考答案:

Spark官方提供了两种方法实现从RDD转换得到DataFrame，第⼀种方法是利用反射机制来推断包含特定类型对象的Schema，这种方式适用于对已知数据结构的RDD转换；（3分）

第二种方法通过编程接口构造⼀个Schema，并将其应用在已知的RDD数据中。 （2分）

11、什么是DStream窗口操作。 （较难）

参考答案: 在DStream流上，将⼀个可配置的长度设置为窗口，以⼀个可配置的速率向前移动窗口。（2分）根据窗口操作，对窗口内的数据进行计算，每次落在窗口内的RDD数据会被聚合起来计算，生成的RDD会作为Window DStream的⼀个RDD。 （3分）

12、 简述说明什么是DStream流。 （较难）

参考答案:

DStream是Spark Streaming提供的⼀个高级抽象流，（2分）表示连续的数据流，（2分）其内部结构是由⼀系列连续的RDD组成。 （1分）

13、 请说明什么是Apache Kafka。 （中）

参考答案:

Kafka是⼀个高吞吐量的分布式发布订阅消息系统（3分），能够将消息从⼀个端点传递到另外⼀个端点。 （2分）

14、 简述Spark的部署方式。 （中）

参考答案:

Spark部署模式分为Local模式（本地单机模式）和集群模式（2分），在Local模式下，常用于本地开发程序与测试，⽽集群模式⼜分为Standalone模式（集群单机模式）、Yarn模式和Mesos模式。 （3分）

15、 简述Scala语言的显著特性。 （中）

参考答案:

Scala语言具有面向对象的、函数式编程的、静态类型的、可扩展的、可以交互操作的。

（每个点1分，答够5个点即可得满分）

16、Spark master使用zookeeper进行HA的，有哪些元数据保存在Zookeeper？ （中）

答：spark通过这个参数spark.deploy.zookeeper.dir指定master元数据在zookeeper中保存的位置，包括Worker，Driver和 Application以及Executors。standby节点要从zk中，获得元数据信息，恢复集群运行状态，才能对外继续提供服务，作业提交资源申请等，在恢复前是不能接受请求的。

（每个点1分，答够5个点即可得满分）

17、 Spark master HA 主从切换过程不会影响集群已有的作业运行，为什么？ （中）

答：因为程序在运行之前，已经申请过资源了（3分），driver和Executors通讯，不需要和master进行通讯的。（2分）

18、 Spark on Mesos中，什么是的粗粒度分配，什么是细粒度分配，各自的优点和缺点是什么？ （中）

答：1）粗粒度：启动时就分配好资源， 程序启动，后续具体使用就使用分配好的资源，不需要再分配资源；好处：作业特别多时，资源复用率高，适合粗粒度；不好：容易资源浪费，假如⼀个job有1000个task，完成了999个，还有⼀个没完成，那么使用粗粒度，999个资源就会闲置在那里，资源浪费。（3分）

2）细粒度分配：用资源的时候分配，用完了就⽴即回收资源，启动会⿇烦⼀点，启动⼀次分配⼀次，会⽐较⿇烦。 （2分）

19、如何配置spark master的HA？ （中）

1)配置zookeeper

2)修改spark\_env.sh⽂件,spark的master参数不在指定，添加如下代码到各个master节点

export SPARK\_DAEMON\_JAVA\_OPTS="-Dspark.deploy.recoveryMode=ZOOKEEPER -

Dspark.deploy.zookeeper.url=zk01:2181,zk02:2181,zk03:2181 -Dspark.deploy.zookeeper.dir=/spark"

3) 将spark\_env.sh分发到各个节点

4)找到⼀个master节点，执行./start-all.sh，会在这里启动主master,其他的master备节点，启动master命令: ./sbin/start-

master.sh

5)提交程序的时候指定master的时候要指定三台master，例如

./spark-shell --master spark://master01:7077,master02:7077,master03:7077

（每个点1分，答够5个点即可得满分）

20、 Apache Spark有哪些常见的稳定版本，Spark1.6.0的数字分别代表什么意思？ （中）

答：常见的大的稳定版本有Spark 1.3,Spark1.6, Spark 2.0 ，Spark1.6.0的数字含义

1）第⼀个数字：1

major version : 代表大版本更新，⼀般都会有⼀些 api 的变化，以及大的优化或是⼀些结构的改变； （1分）

2）第⼆个数字：6

minor version : 代表⼩版本更新，⼀般会新加 api，或者是对当前的 api 就行优化，或者是其他内容的更新，⽐如说 WEB UI的更新等等； （2分）

3）第三个数字：0

patch version ， 代表修复当前⼩版本存在的⼀些 bug，基本不会有任何 api 的改变和功能更新； （2分）

21.driver的功能是什么？ （中）

答：

1）⼀个Spark作业运行时包括⼀个Driver进程，也是作业的主进程，具有main函数，并且有SparkContext的实例，是程序的⼈⼝点；（2分）

2）功能：负责向集群申请资源，向master注册信息，负责了作业的调度，，负责作业的解析、生成Stage并调 度Task到Executor上。包括DAGScheduler，TaskScheduler。 （3分）

22.spark的有⼏种部署模式，每种模式特点？ （中）

1）本地模式 （1分）

Spark不⼀定⾮要跑在hadoop集群，可以在本地，起多个线程的方式来指定。将Spark应用以多线程的方式直接运行在本地，

⼀般都是为了方便调试，本地模式分三类

• local：只启动⼀个executor

• local[k]:启动k个executor

• local[\*]：启动跟cpu数目相同的 executor

2)standalone模式 （1分）

分布式部署集群， 自带完整的服务，资源管理和任务监控是Spark自己监控，这个模式也是其他模式的基础，

1. Spark on yarn模式（2分）

分布式部署集群，资源和任务监控交给yarn管理，但是目前仅支持粗粒度资源分配方式，包含cluster和client运行模 式，cluster适合生产，driver运行在集群子节点，具有容错功能，client适合调试，dirver运行在客户端

1. Spark On Mesos模式。（1分）

官方推荐这种模式（当然，原因之⼀是血缘关系）。正是由于Spark开发之初就考虑到支持 Mesos，因此，目前⽽言，Spark运行在Mesos上会⽐运行在YARN上更加灵活，更加自然。

23.Spark技术栈有哪些组件，每个组件都有什么功能，适合什么应用场景？ （中）

答：可以画⼀个这样的技术栈图先，然后分别解释下每个组件的功能和场景

1）Spark core：是其它组件的基础，spark的内核，主要包含：有向循环图、RDD、Lingage、Cache、broadcast等，并封装

了底层通讯框架，是Spark的基础。

2）SparkStreaming是⼀个对实时数据流进行高通量、容错处理的流式处理系统，可以对多种数据源（如Kdfka、Flume、Twitter、Zero和TCP 套接字）进行类似Map、Reduce和Join等复杂操作，将流式计算分解成⼀系列短⼩的批处理作业。

3）Spark sql：Shark是SparkSQL的前身，Spark SQL的⼀个重要特点是其能够统⼀处理关系表和RDD，使得开发⼈员可以

轻松地使用SQL命令进行外部查询，同时进行更复杂的数据分析

4）BlinkDB ：是⼀个用于在海量数据上运行交互式 SQL 查询的大规模并行查询引擎，它允许用户通过权衡数据精度来提升查

询响应时间，其数据的精度被控制在允许的误差范围内。

5）MLBase是Spark生态圈的⼀部分专注于机器学习，让机器学习的门槛更低，让⼀些可能并不了解机器学习的用户也能方便

地使用MLbase。MLBase分为四部分：MLlib、MLI、ML Optimizer和MLRuntime。

1. GraphX是Spark中用于图和图并行计算

（每个点1分，答够5个点即可得满分）

24.Spark中Work的主要工作是什么？ （中）

答：主要功能：管理当前节点内存，CPU的使用状况，接收master分配过来的资源指令，通过ExecutorRunner启动程序分配 任务，worker就类似于包工头，管理分配新进程，做计算的服务，相当于process服务。（1分）

需要注意的是：1）worker会不会汇 报当前信息给master，worker⼼跳给master主要只有workid，它不会发送资源信息以⼼跳的方式给mater，master分配的时候 就知道work，只有出现故障的时候才会发送资源。（2分）

2）worker不会运行代码，具体运行的是Executor是可以运行具体 appliaction写的业务逻辑代码，操作代码的节点，它不会运行程序的代码的。 （2分）

25.Spark为什么⽐mapreduce快？ （中）

答：1）基于内存计算，减少低效的磁盘交互；（2分）

1. 高效的调度算法，基于DAG；（2分）

3)容错机制Linage，精华部分就是DAG和Lingae （1分）

26、 什么是Spark（中）

Spark基于内存的，（2分）用于大规模数据处理（离线计算、实时计算、快速查询（交互式查询））的统一分析引擎。（3分）

27、Spark特点（中）

快：Spark计算速度是MapReduce计算速度的10-100倍（1分）

易用：（算法多）MR支持1种计算模型，Spsark支持更多的计算模型。（1分）

通用：Spark 能够进行离线计算、交互式查询（快速查询）、实时计算、机器学习、图计算等（2分）

兼容性：Spark支持大数据中的Yarn调度，支持mesos。可以处理hadoop计算的数据。（1分）

28、Spark运行模式（中）

1.local本地模式(单机)–开发测试使用

2.standalone独立集群模式–开发测试使用

3.standalone-HA高可用模式–生产环境使用

4.on yarn集群模式–生产环境使用

5.on cloud集群模式–中小公司未来会更多的使用云服务

（每个点1分，答够5个点即可得满分）

29、Spark编写代码流程是什么？（中）

1、创建一个 Sparkconf对象，设置app名称（1分）

2、创建一个SparkContext,设置日志级别（1分）

3、读取数据，对数据进行计算（2分）

4、保存数据（1分）

30、什么是RDD（易）

弹性分布式数据集（数据存储在内存）（2分），一个不可变、可分区、里面的元素可并行计算的集合（3分）

31、RDD的主要属性（中）

1、数据集的基本组成单位是一组分片(Partition)或一个分区(Partition)列表

每个分片都会被一个计算任务处理，分片数决定并行度。（2分）

2、一个函数会被作用在每一个分区。（2分）

3、一个RDD会依赖于其他多个RDD，RDD的每次转换都会生成一个新的RDD（1分）

32、RDD的算子分为两类，分别是什么？（中）

1.Transformation转换操作:返回一个新的RDD（2分）

2.Action动作操作:返回值不是RDD惰性计算，遇到Transformation不计算，遇到Action在真正计算。（3分）

33、Rdd数据持久化什么作用？（中）

1、对多次使用的rdd进行缓存，缓存到内存，当后续频繁使用时直接在内存中读取缓存的数据，不需要重新计算。 (Persist、Cache)）（2分）

2、将RDD结果写入硬盘（容错机制），当RDD丢失数据时，或依赖的RDD丢失数据时，可以使用持久化到硬盘的数据恢复。（3分）

34、cache和Checkpoint的区别是什么（中）

位置（2分）

Persist 和 Cache将数据保存在内存

Checkpoint将数据保存在HDFS

生命周期（2分）

Persist 和 Cache 程序结束后会被清除或手动调用unpersist方法。

Checkpoint永久存储不会被删除。

RDD依赖关系（血统Lineage）（1分）

Persist和Cache，不会丢掉RDD间的依赖链/依赖关系

Checkpoint会斩断依赖链

35、什么是宽窄依赖？（中）

窄依赖:父RDD的一个分区只会被子RDD的一个分区依赖（2分）

宽依赖:父RDD的一个分区会被子RDD的多个分区依赖(涉及到shuffle)（3分）

36、什么是DAG（中）

DAG：指的是数据转换执行的过程（2分），有方向，无闭环(其实就是RDD执行的流程)（3分）

37、DAG边界是什么？（中）

开始:通过SparkContext创建的RDD

结束:触发Action，一旦触发Action就形成了一个完整的DAG（2分）

一个Spark应用中可以有一到多个DAG，取决于触发了多少次Action（1分）

一个DAG中会有不同的阶段/stage，划分阶段/stage的依据就是宽依赖（1分）

一个阶段/stage中可以有多个Task，一个分区对应一个Task（1分）

38、什么是Spark中的Application？（中）

Application指的是用户编写的Spark应用程序/代码，（2分）包含了Driver功能代码和分布在集群中多个节点上运行的Executor代码。（3分）

39、什么是Spark中的Driver？（中）

Spark中的Driver即运行上述Application的Main()函数并且创建SparkContext，（2分）SparkContext负责和ClusterManager通信，进行资源的申请、任务的分配和监控等（3分）

40、什么是Spark中的Cluster Manager？（较难）

Cluster Manager指的是在集群上获取资源的外部服务，（2分）Standalone模式下由Master负责，Yarn模式下ResourceManager负责;（3分）

41、Spark执行任务的基本流程（较难）

1) Spark应用被提交–>SparkContext向资源管理器注册并申请资源 (??) -->启动Executor（2分）

2) RDD–>构建DAG–>DAGScheduler划分Stage形成TaskSet–>TaskScheduler提交Task–>Worker上的Executor执行Task（3分）

42、广播变量的作用（较难）

在每个机器上缓存一份、不可变的、只读的、相同的变量，该节点每个任务都能访问。(3分)起到节省资源的作用，和优化的所用。（2分）

43、什么是SparkSQL?SparkSQL底层的数据抽象类型是什么?（较难）

用于处理结构化数据的Spark模块。(3分)

DataFrame和DataSet（2分）

44、请简述Hive和SparkSQL的对比（较难）

Hive是将sql转化成MapReduce进行计算（降低学习成本、提高开发效率）（2分）

SparkSQL是将sql转化成rdd集进行计算（降低学习成本、提高开发效率）（3分）

45、什么是DataFrame？？（中）

DataFrame是以RDD为基础的带有Schema元信息的分布式数据集。（5分，根据答题内容酌情删减）

46、什么是DataSet??（中）

含有类型信息的DataFrame就是DataSet（DataSaet=DataFrame+类型= Schema+RDD\*n+类型）（5分，根据答题内容酌情删减）

47、SparkSQL查询数据的形态有哪些？请举例说明。（中）

1、类似方法调用，领域特定语言(DSL)。

personDF.select(" i d " , "id","id",“name”," a g e " + 1 ) . f i l t e r ( "age"+1).filter("age"+1).filter(“age”>25).show（）（3分）

2、SQL语句

spark.sql(“select \* from personDFT where age >25”).show（）（2分）

48、添加Schema的方式分几种，分别是什么？（中）

第1种：指定列名添加Schema（1分）

第2种：通过StructType指定Schema（2分）

第3种：编写样例类，利用反射机制推断Schema（2分）

1. 指定列名添加Schema代码流程？（中）

答：

1 创建sparksession

2 创建sc，读取数据并加工

3 设置表结构 ttRDD.toDF(“id”,“name”,“age”)

4 注册成表并查询

5 关闭sc sparksession

（每个点1分，答够5个点即可得满分）

50、通过StructType指定Schema代码流程是什么？（中）

1 创建sparksession

2 创建sc

3 读取数据并加工

4 设置表结构

5 创建DS、DF

6 注册成表并查询

7 关闭sc、sparksession

（每个点1分，答够5个点即可得满分）

51、利用反射机制推断Schema代码流程是什么（中）

1 创建sparksession

2 创建sc

3 读取数据并加工

val PersonRDD: RDD[Person] = ttRDD.map(z=>Person(z(0).toInt,z(1),z(2).toInt))

4 RDD转DF

val personDF: DataFrame = PersonRDD.toDF()

5 注册成表并查询

6 关闭sc、sparksession

（每个点1分，答够5个点即可得满分）

52、RDD、DF、DS三者之间的转化方法是什么，请举例说明？（中）

转换成RDD .rdd

转换成DF .toDF()

转换成DS

RDD->DS .toDS()

DF->DS .as[Person]

（每个点1分，答够5个点即可得满分）

53、什么是Spark Streaming？什么是DStream？（中）

Spark Streaming是一个基于Spark Core之上的实时计算框架（2分）

代表持续性的输入的数据流和经过各种Spark算子操作后的输出的结果数据流。本质上就是按照时间间隔划分成一批一批的连续的RDD（3分）

54、阐明RDD、DataFrame、DataSet、DStream数据抽象之间的关系。（中）

DStream=RDD1(t1)+ RDD2(t2)+ RDD3(t3)+ RDD4(t4)+….（1分）

DataSet = DataFrame+类型 = RDD+结构+类型（2分）

DataFrame = RDD+结构（2分）

55、Spark Streaming代码过程（中）

1 创建sparkConf

2 创建一个sparkcontext

3 创建streamingcontext

4 接收数据并根据业务逻辑进行计算

5 开启计算任务

6 等待关闭

（每个点1分，答够5个点即可得满分）

56、什么是Structured Streaming？（中）

答：Structured Streaming是一个基于Spark SQL引擎的可扩展（2分）、容错的（2分）流处理引擎（1分）

（总分5分，根据回答内容酌情删减分数）

57、Structured Streaming模型特点有哪些？（中）

答：是一个不断增长的动态表格，（2分）新数据被持续不断地添加到表格的末尾（2分）对动态数据源进行实时查询，就是对当前的表格内容执行一次 SQL 查询。（1分）

（总分5分，根据回答内容酌情删减分数）

58、Structured Streaming应用场景（中）

答：将数据源映射为类似于关系数据库中的表，（2分）（SparkSQL中的DF/DS）然后将经过计算得到的结果映射为另一张表.（3分）

（总分5分，根据回答内容酌情删减分数）

59.简单说⼀下hadoop和spark的shuffle相同和差异？ （中）

答：1）从 high-level 的角度来看，两者并没有大的差别。都是将 mapper（

Spark 里是 ShuffleMapTask）的输出进行 partition，不同的 partition 送到不同的 reducer（Spark 里 reducer 可能是下⼀个 stage 里的 ShuffleMapTask，也可能是 ResultTask）。Reducer 以内存作缓冲区，边 shuffle 边 aggregate 数据，等到数据 aggregate 好以后进行 reduce() （Spark 里可能是后续的⼀系列操作）。 （2分）

2）从 low-level 的角度来看，两者差别不⼩。 Hadoop MapReduce 是 sort-based，进⼊ combine() 和 reduce() 的 records 必须先 sort。这样的好处在于 combine/reduce() 可以处理大规模的数据，因为其输⼊数据可以通过外排得到（mapper 对每段数据先做排序，reducer 的 shuffle 对排好序的每段数据做归并）。目前的 Spark 默认选择的是 hash-based，通常使用HashMap 来对 shuffle 来的数据进行 aggregate，不会对数据进行提前排序。 （3分）

60、Mapreduce和Spark的都是并行计算，那么他们有什么相同和区别 （中）

答：两者都是用mr模型来进行并行计算:

1)hadoop的⼀个作业称为job，job里面分为map task和reduce task，每个task都是在自己的进程中运行的，当task结束时，进程也会结束。 （2分）

2)spark用户提交的任务成为application，⼀个application对应⼀个sparkcontext，app中存在多个job，每触发⼀次action操作就会产生⼀个job。这些job可以并行或串行执行，每个job中有多个stage，stage是shuffle过程中DAGSchaduler通过RDD之间的依赖关系划分job⽽来的，每个stage里面有多个task，组成taskset有TaskSchaduler分发到各个executor中执行，executor 的生命周期是和app⼀样的，即使没有job运行也是存在的，所以task可以快速启动读取内存进行计算。 （2分）

3)hadoop的job只有map和reduce操作，表达能⼒⽐较⽋缺⽽且在mr过程中会重复的读写hdfs，造成大量的io操作，多个job需 要自己管理关系。

spark的迭代计算都是在内存中进行的，API中提供了大量的RDD操作如join，groupby等，⽽且通过DAG图可以实现良好的容错。（1分）

61.简述RDD机制？ （中）

答：rdd分布式弹性数据集，简单的理解成⼀种数据结构，是spark框架上的通用货币。

所有算子都是基于rdd来执行的，不同的场景会有不同的rdd实现类，但是都可以进行互相转换。 （3分）

rdd执行过程中会形成dag图，然后形成lineage保证容错性等。 从物理的角度来看rdd存储的是block和node之间的映射。 （2分）

62、spark有哪些组件？ （中）

答：主要有如下组件：

1）master：管理集群和节点，不参与计算。

2）worker：计算节点，进程本身不参与计算，和master汇报。

3）Driver：运行程序的main方法，创建spark context对象。

4）spark context：控制整个application的生命周期，包括dagsheduler和task scheduler等组件。

5）client：用户提交程序的⼊⼝。

（每个点1分，答够5个点即可得满分）

63、spark工作机制？ （中）

答：用户在client端提交作业后，会由Driver运行main方法并创建spark context上下⽂。 （2分）

执行add算子，形成dag图输⼊dagscheduler，按照add之间的依赖关系划分stage输⼊task scheduler。 task scheduler会将 stage划分为task set分发到各个节点的executor中执行。 （3分）

64、spark的优化怎么做？ （难）

答： spark调优⽐较复杂，但是大体可以分为三个方面来进行，1）平台层面的调优：防⽌不必要的jar包分发，提高数据的本 地性，选择高效的存储格式如parquet，（2分）2）应用程序层面的调优：过滤操作符的优化降低过多⼩任务，降低单条记录的资源开 销，处理数据倾斜，复用RDD进行缓存，作业并行化执行等等，（2分）3）JVM层面的调优：设置合适的资源量，设置合理的JVM，启用高效的序列化方法如kyro，增大off head内存等等 （1分）

65.简要描述Spark分布式集群搭建的步骤 ？（难）

1）准备linux环境，设置集群搭建账号和用户组，设置ssh，关闭防⽕墙，关闭seLinux，配置host，hostname

2）配置jdk到环境变量

3）搭建hadoop集群，如果要做master ha，需要搭建zookeeper集群

修改hdfs-site.xml,hadoop\_env.sh,yarn-site.xml,slaves等配置⽂件

4）启动hadoop集群，启动前要格式化namenode

5）配置spark集群，修改spark-env.xml，slaves等配置⽂件，拷贝hadoop相关配置到spark conf目录下

6)启动spark集群。

（每个点1分，答够5个点即可得满分）

66.什么是RDD宽依赖和窄依赖？ （难）

RDD和它依赖的parent RDD(s)的关系有两种不同的类型，即窄依赖（narrow dependency）和宽依赖（wide dependency）。 （3分）

1）窄依赖指的是每⼀个parent RDD的Partition最多被子RDD的⼀个Partition使用 （1分）

2）宽依赖指的是多个子RDD的Partition会依赖同⼀个parent RDD的Partition （1分）

67.spark-submit的时候如何引⼊外部jar包？ （难）

方法⼀：spark-submit –jars

根据spark官⽹，在提交任务的时候指定–jars，用逗号分开。

命令：spark-submit --master yarn-client --jars .jar,.jar （2分）

方法二：extraClassPath

提交时在spark-default中设定参数，将所有需要的jar包考到⼀个文件里，然后在参数中指定该目录就可以了 （3分）

68.Spark缓存中的cache和pesist的区别是说明，请简要描述？（难）

答：1）cache和persist都是用于将⼀个RDD进行缓存的，这样在之后使用的过程中就不需要重新计算了，可以大大节省程序运行时间；(2分)

2） cache只有⼀个默认的缓存级别MEMORY\_ONLY ，cache调用了persist，⽽persist可以根据情况设置其它的缓存级别；(2分)

3）executor执行的时候，默认60%做cache，40%做task操作，persist最根本的函数，最底层的函数(1分)

69、Catalyst优化器内部包含五大组件，分别是什么？（难）

答：1）、SqlParse （1分）

2）、Analyze （1分）

3）、Optimizer （1分）

4）、Planner（1分）

5）、CostModel （1分）

70、HBase表的列是由哪些字段组成。 （中）

答：1）列族名 （2分）

2）限定符 （2分）

3）列名 （1分）

71、请描述HBase集群的特点？ （中）

答：1）、HBase集群中有两个角⾊ （1分）

2）、HRegionServer是HBase集群的从节点 （1分）

3）、HMaster是HBase的主节点 （1分）

4）、HBase通过zookeeper实现主从架构（2分）

72、Spark生态圈的组件有哪些，请列举。 （中）

答：1）、Spark SQL （1分）

2）、Spark Streaming （1分）

3）、GraphX （1分）

4）、MLlib （1分）

5）、SparkRDD（1分）

73、请列举Scala的数据结构？ （中）

答：1）、数组 （1分）

2）、元组 （2分）

3）、集合 （2分）

74、Spark SQL作为分布式SQL查询引擎，让⽤户可以通过哪些方式实现对结构化数据的处理？（中）

答：1）、SQL （1分）

2）、DataFrames API（2分）

3）、Datasets API （2分）

75、Spark编程支持的语言有哪些？（中）

答：Spark编程支持Java、Python、Scala或R语言。 （每个语言1分，回答上四种给满分）

76、Spark RDD⾏动算⼦有哪些？ （中）

答：行动算子有foreach(func) 、collect() 、take(n) 、count、countByKey

Reduce（每个算子1分，不局限于上述算子，回答满5个给满分）

77、Kafka的特点有哪些？（中）

答：Kafka使⽤消费组（Consumer Group）的概念统⼀了点对点消息传递模式和发布订阅消息传递模式（2分），当Kafka使⽤点对点模式时，它可以将处理工作任务平均分配给消费组中的消费者成员；当使⽤发布订阅模式时，它可以将消息⼴播给多个消费组。（2分）Kafka采⽤多个消费组结合多个消费者，既可以扩展消息处理的能⼒，也允许消息被多个消费组订阅。（1分）

1. 请列出 Scala 支持的数据类型？（中）

答： Scala 支持的数据类型有int、Long、Float、Double、Char、String等

（每个类型1分，不局限于上述类型，回答满5个给满分）

## 79、请描述Scala语言特点?（中）

答：(1) Scala是面向对象的语言。（1分）

(2) Scala是函数式编程的语言。（1分）

(3) Scala是静态类型的。（1分）

(4) Scala是可扩展的。（1分）

(5) Scala是可以交互操作的。（1分）

80、Scala的控制结构语句有哪几种？分别是什么？（中）

答：在Scala中，控制结构语句包括条件分支语句和循环语句（2分）。其中，条件分支语句有if语句、if...else语句、if...else if...else语句以及if...else嵌套语句;（2分）循环语句有for循环，while循环和do...while循环。（1分）