关于 lab3 说明.

因为 Lab3-contest 里面是 EquiJoin,所以我把 Join 代码替换为 HashEquiJoin,做完 contest 之后忘了改回去,因此 lab3 提交的 tar 无法通过 Test. 在 lab4 实验的时候发现了想起来了,现在已经修改回去,但是会使得助教运行 Lab3-contest 表现不好,不过 Contest 结果已经卸载 lab3 报告里面.麻烦助教了.

Describe any design decisions you made.

EX1

分情况讨论

处理特殊情况,比如 取值超过最大值,或者取值小于最小值

EQUALS:
    estimateTuples = _histogram[bucketIndex] / _bucketSize;
NOT_EQUALS:
    estimateTuples = _numTuples - _histogram[bucketIndex] / _bucketSize;
GREATER_THAN:
    大于最大值 0 小于最大值 1 ,其余情况,按比例等分
GREATER_THAN_OR_EQ:
LESS_THAN:
LESS_THAN_OR_EQ:
    与 GREATER_THAN 类似

EX2

estimateSelectivity（int field，Predicate.Op op，Field constant）：估计表中谓词字段 op 常量的选择性。
estimateScanCost（）：此方法会估计顺序扫描文件的成本。可以假定没有任何搜索，也没有页面在缓冲池中。此方法可能会使用在构造函数中计算的成本或大小。
estimateTableCardinality（double selectivityFactor）：此方法返回关系中元组的数量，前提是应用了具有选择性 selectiveFactor 的谓词。此方法可能会使用在构造函数中计算的成本或大小。

EX3

1.对于等式连接，当其中一个属性是主键时，连接产生的元组数量不能大于非主键属性的基数。

2.对于没有主键的等式连接，很难说输出大小是多少 - 它可能是表的基数乘积的大小（如果两个表对所有元组具有相同的值） - 或者它可以是 0.构建一个简单的启发式（比如两个表中较大的表的大小）就可以。
3.对于范围扫描，同样很难说出关于尺寸的任何准确信息。输出的大小应该与输入的大小成正比。假定交叉产品的固定比例是通过范围扫描（例如 30%）发射的，这很好。通常，范围连接的成本应该大于两个相同大小的表的非主键等同连接的成本。

EX4

该方法应该对 joins 类成员进行操作，并返回一个新的 Vector 来指定应该完成联接的顺序。此向量的项 0 表示左深度计划中最左边最底部的连接。返回的向量中的相邻连接应共享至少一个字段，以确保该计划处于深度。这里 stats 是一个对象，可以查找出现在查询的 FROM 列表中的给定表名称 的 TableStats。
filterSelectivities 允许查找表上任何谓词的选择性; 保证在 FROM 列表中每个表名有一个条目 。

Discuss and justify any changes you made to the API.

添加一些必要的 private 成员

Describe any missing or incomplete elements of your code.

缺少一些非法性检测,需要保证,使用者完全按照接口使用.

Describe how long you spent on the lab, and whether there was anything you found particularly difficult or confusing.

花费 大概 20 个小时
难以把握整个 simpleDB 各个类,各个函数直间的调用关系
光看 api 不能够具体知道整个流程是什么,希望之后能够给出流程图,或者整个 simpledp 的类图