COP 6726 – Database System Implementation

Project 4_1: Statistical Estimation

Group Members:

Sanjay Reddy Banda, UF ID: 5878-2239

Suprith Reddy Gurudu, UF ID: 9961-2134

Compilation and Execution:

Bin files are generated by a2-test.cc, run the following command:

Compile:

```
>> make a2test.out
```

Run:

```
>> ./a2test.out
```

And follow necessary instructions on the screen.

To compile the code, run the following command:

```
>> make
```

To execute the test.cc code, change the directory to the specific folder (a4-1test) and run the following command:

```
>> ./test.out <query(0-11)>
```

To compile the gTest (gtests.cc) code, run the following command:

```
>> make gtest.out
```

To execute the gTest (gtests.cc) code, run the following command:

```
>> ./gtest.out
```

Code Explanation (modified methods):

Filename: Statistics.cc

Classname: Attribute

Methods:

Attribute(int num, string name):

Constructor to initialize attribute with name and unique tuples.

Attribute(const Attribute &copyMe):

Copy Constructor to perform deep copy of the attribute object.

Attribute &operator = (const Attribute &copyMe):

Overloading equals to operand to perform deep copy of the attribute object.

Classname: Relation

Methods:

Relation(int num,string name):

Constructor to initialize relation with name and unique tuples.

Relation(const Relation &copyMe):

Copy Constructor to perform deep copy of the relation object.

Relation &operator = (const Relation &copyMe):

Overloading equals to operand to perform deep copy of the relation object.

bool isRelationPresent (string name):

Checks and returns if there exists a given relation or not.

Classname: Statistics

Methods:

Statistics(Statistics &copyMe):

Copy Constructor to perform deep copy of the statistics object.

Statistics &operator= (Statistics &copyMe):

Overloading equals to operand to perform deep copy of the statistics object.

*int GetRelationForOperand(Operand *op,char *relationName[],int numJoin,Relation &relationInfo):*

Returns 0 if there exists a relation and copies the relation object to relationInfo else return -1.

*double OrOperand(OrList *orList,char *relationName[],int numJoin):*

calculates the selectivity of the given OrList.

*double AndOperand(AndList *andList,char *relationName[],int numJoin):*

calculates the selectivity of the given AndList.

*double CompOperand(ComparisonOp *compOp,char *relationName[],int numJoin):*

calculates the selectivity of the given Comparison operand.

*void AddRel(char *relName, int numTuples):*

Adds the relation to the current statistics object.

*void AddAtt(char *relName, char *attName, int numDistincts):*

Adds the attribute to the given relation in the current statistics object.

*void CopyRel(char *oldName, char *newName):*

Makes a copy of the relation object with the new name.

*void Read(char *fromWhere):*

Reads the data from the file and modifies the current statistics object.

*void Write(char *fromWhere):*

Writes the current statistics object into a file.

*void Apply(struct AndList *parseTree, char *relNames[], int numToJoin):*

Modifies the statistics object after applying the given cnf.

*double Estimate(struct AndList *parseTree, char **relNames, int numToJoin):*

Estimates the result record count after applying given cnf.

Filename: gtest.cc

```
TEST(STATISTICS, TestCase1) –
      Google test for validating the test case for CNF => (l_returnflag
= 'R') AND (l_discount < 0.04 OR l_shipmode = 'MAIL') scenario. It
verifies by total number of tuples estimated by the function.
TEST(STATISTICS, TestCase2) –
      Google test for validating the test case for CNF => (c_custkey =
o_custkey) scenario. It verifies by total number of tuples estimated by
the function.
TEST(STATISTICS, TestCase3) –
      Google test for validating the test case for CNF => (c_mktsegment
= 'BUILDING') AND (c_custkey = o_custkey) AND (o_orderdate < '1995-03-
1') scenario. It verifies by total number of tuples estimated by the
function.
TEST(STATISTICS, TestCase4) –
      Google test for validating the test case for CNF => (c_custkey =
o_custkey)  AND (o_orderdate > '1994-01-23') scenario. It verifies by
total number of tuples estimated by the function.
```

Description for the format of statistics.txt:

Line 1: Identifies total number of relations in the statistics object.

For each relation:

Line 2: Relation Name

Line 3: Total number of estimated tuples

Line 4: isJoint present

If Line 4 == 1:

Line 5: number of Joints

For each Joint:

Line 6: Joint name (Relation name)

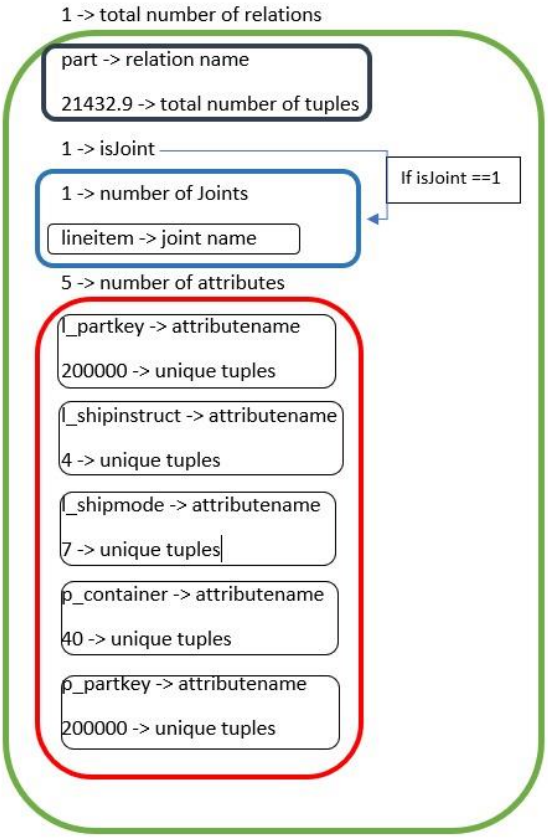
Line 7: number of attributes in the statistics object

For each attribute:

Line 8: Attribute Name

Line 9: Unique tuples of the attribute

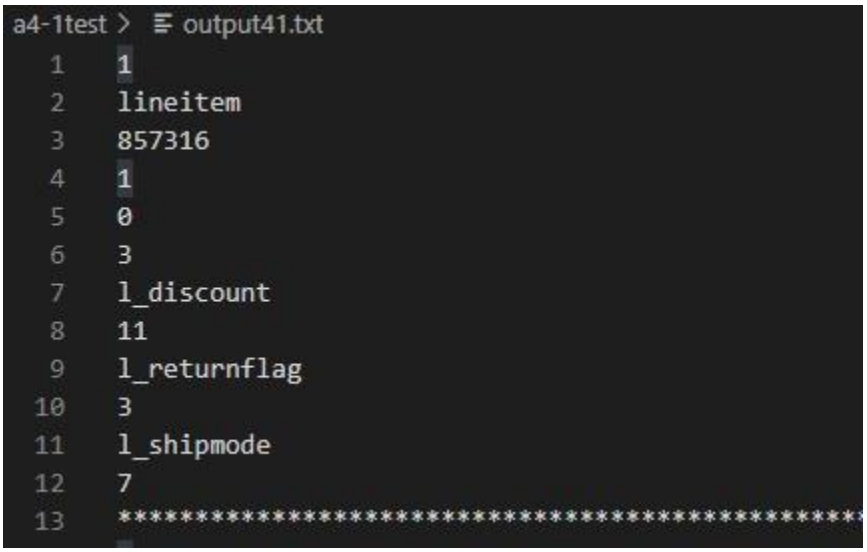
Please check the below pictorial representation:



Results for the Test Cases from output41.txt:

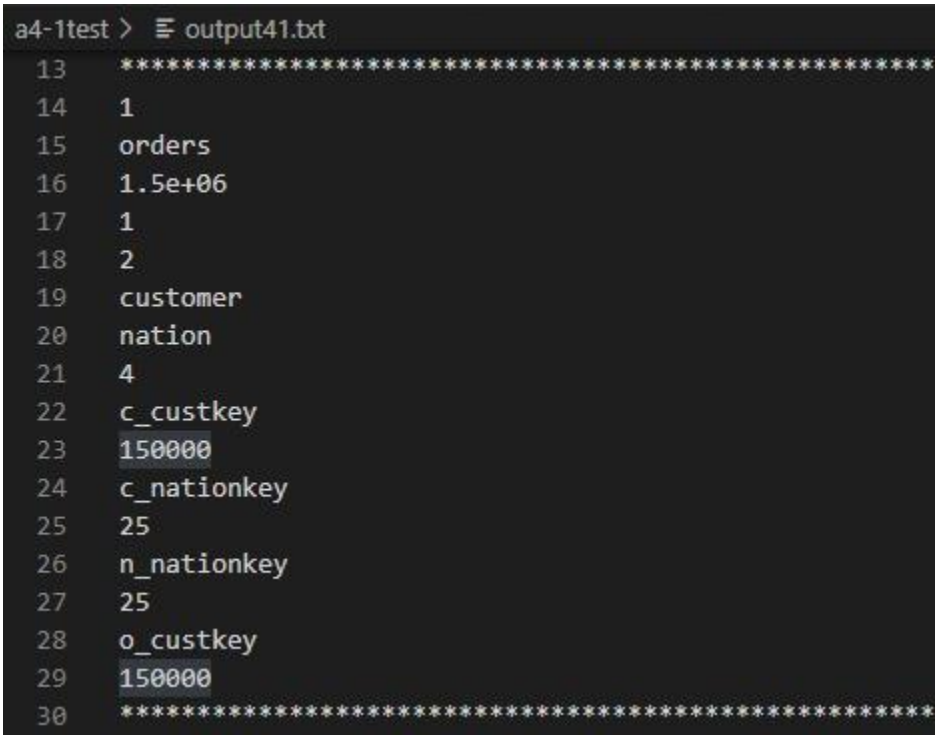
Test Case 1:

Input - `CNF => (l_returnflag = 'R') AND (l_discount < 0.04 OR l_shipmode = 'MAIL')`



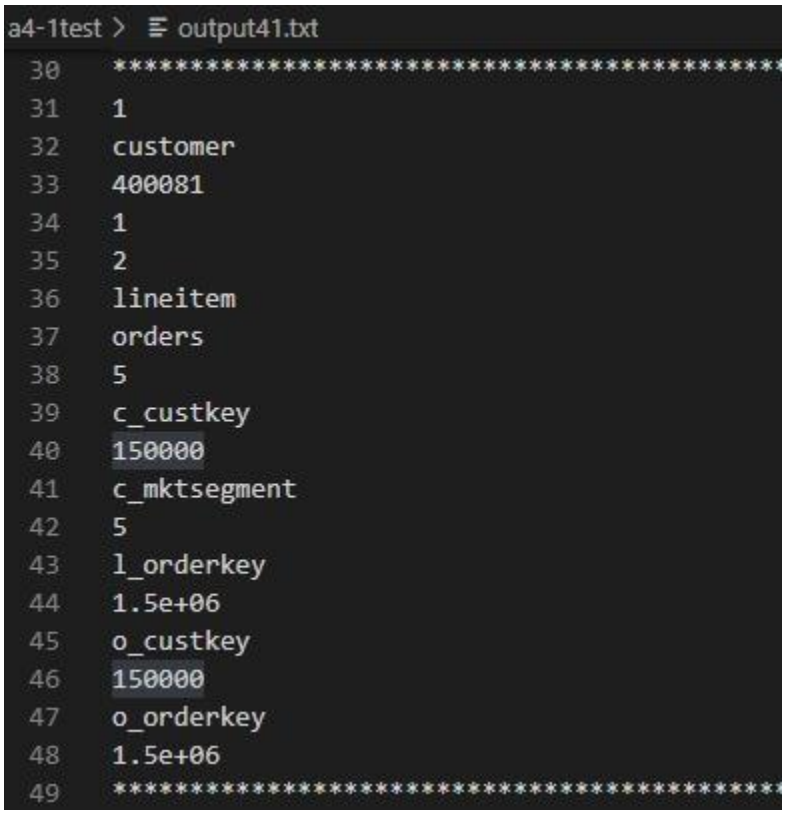
Test Case 2:

Input - `(c_custkey = o_custkey)`



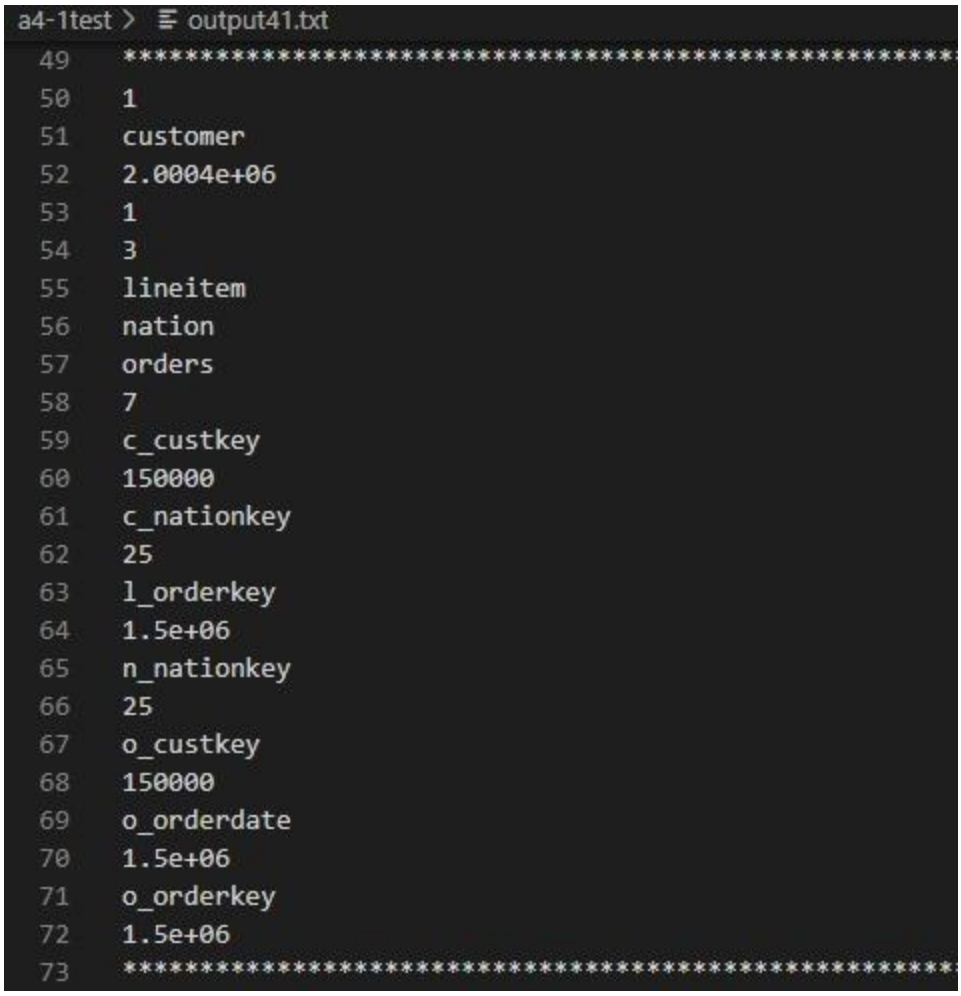
Test Case 3:

Input - `(c_mktsegment = 'BUILDING') AND (c_custkey = o_custkey) AND (o_orderdate < '1995-03-1')`



Test Case 4:

Input - `(c_custkey = o_custkey) AND (o_orderdate > '1994-01-23')`


a4-1test > output41.txt
49 **********
50 1
51 customer
52 2.0004e+06
53 1
54 3
55 lineitem
56 nation
57 orders
58 7
59 c_custkey
60 150000
61 c_nationkey
62 25
63 l_orderkey
64 1.5e+06
65 n_nationkey
66 25
67 o_custkey
68 150000
69 o_orderdate
70 1.5e+06
71 o_orderkey
72 1.5e+06
73 **********

Test Case 5:

Input - (l_partkey = p_partkey) AND (l_shipmode = 'AIR' OR l_shipmode = 'AIR REG') AND (p_container ='SM BOX' OR p_container = 'SM PACK') AND (l_shipinstruct = 'DELIVER IN PERSON')

a4-1test > output41.txt

```
73  ********************************************
74  1
75  part
76  21432.9
77  1
78  1
79  lineitem
80  5
81  l_partkey
82  200000
83  l_shipinstruct
84  4
85  l_shipmode
86  7
87  p_container
88  40
89  p_partkey
90  200000
91  ********************************************
92
```

Results for runTestCases.sh:

```
sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a4-1test$ make
g++ -O2 -Wno-deprecated  -g -c Statistics.cc
g++ -O2 -Wno-deprecated  -o a4-1.out Record.o Comparison.o ComparisonEngine.o Schema.o File.o DBFile.o Pipe.o BigQ.o Statistics.o y.tab.o lex.yy.o test.o -lfl -lpthread
sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a4-1test$ sh runTestCases.sh
Q1: Estimation: 857316


Q2: Estimation: 1.5e+06


Q5: Estimation: 400081


Q10: Estimation: 2.0004e+06


Q11: Estimation: 21432.9


sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a4-1test$
```

Results for gTests:

```
sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a4-1test$ ./gtest.out
[==========] Running 4 tests from 1 test suite.
[----------] Global test environment set-up.
[----------] 4 tests from STATISTICS
[ RUN      ] STATISTICS.TestCase1
[       OK ] STATISTICS.TestCase1 (0 ms)
[ RUN      ] STATISTICS.TestCase2
[       OK ] STATISTICS.TestCase2 (0 ms)
[ RUN      ] STATISTICS.TestCase3
[       OK ] STATISTICS.TestCase3 (0 ms)
[ RUN      ] STATISTICS.TestCase4
[       OK ] STATISTICS.TestCase4 (0 ms)
[----------] 4 tests from STATISTICS (1 ms total)

[----------] Global test environment tear-down
[==========] 4 tests from 1 test suite ran. (3 ms total)
[  PASSED  ] 4 tests.
sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a4-1test$
```