COP 6726:DataBase System Implementation

Project 4:Part 1:Statistical estimation

Venkateswarlu Tanneru
vtanneru@ufl.edu

Rohit Yerramsetty
rohityerramsetty@ufl.edu

Wednesday 20 April, 2022

1 Instructions to run the Project:

This program was written, run, and tested on the Linux operating system. Please change all of the paths in the "test.cat" file before proceeding with the following steps. The first line of test.cat should provide the path to the catlogue file (catalog path), the path to heap files (dbfile dir), and the path to the tpch-dbgen data tables (tpch-dbgen path) (tpch dir).

- Run the **runtestcases.sh** file to run all the queries at a time and make their results appeared in **outpu.txt** file.
  - make clean
  - make
  - ./runtestcases.out
- Run the **gtest** files to generate gtest results
  - make gtest
  - ./gtest.out

2 Files Location:

- **MakeFile**: Make File runs the file structure in making the interpreted files in project.
- **tpch/**: Contains .tbl files which were generated using tpch data generator.
- **gtest/**: Google Test Unit testing library.
- **catalog**: Catalog file for the schema of .tbl files
- **test/**: Contains .bin files.

3 DBFiles extended function classes:

3.1 Statistics Classes

The Attrib_Info Function ,Rel_Info Function and Statistics funtions are employed for various functioning in the project.

3.2 Attrib_Info :: Attrib_Info (string name, int num)

Constructor that sets the number of distinct tuples and the name of the attribute.

3.3 Attrib_Info :: Attrib_Info (const Attrib_Info& copyMe):

Constructor for deep copying given attribute information into a new one.

3.4 Attrib_Info &Attrib_Info :: operator= (const Attrib_Info& copyMe) :

Constructor that both initializes the attribute and returns the associated operator.

3.5 Rel_Info :: Rel_Info (const Rel_Info& copyMe) :

Constructor for deep copying given relation information into a new one.

3.6 Rel_Info :: Rel_Info (string name, int tuples) :

Constructor that initializes the relation information with the given name and the number of tuples.

3.7 Rel_Info &Rel_Info :: operator= (const Rel_Info& copyMe) :

Constructor that creates the relation and returns the operator that corresponds to it.

3.8 void Statistics_Data :: AddAtt(char* relName, char* attrName, int numDistincts:

This function initializes and adds an attribute to the relationMap.

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

This function initializes and adds to relationMap a relation with the number of tuples specified.

3.10 void Statistics_Data :: CopyRel (char* oldName, char* newName :

The attributes and information are deep copied with a new name in this function.

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

To estimate the final result, we multiply the number of tuples by the cross product number and scale by a factor. In addition, we divide the And List into all of the OrLists, calculate the selectivity factor for each OrList, and multiply them.

3.12 int Statistics_Data :: GetRelForOp(Operand* operand, char* relName[], int numJoin, Rel_Info& relInfo) :

This function finds a relationship for a given operand and returns 0 if it is found, otherwise -1.

3.13 void Statistics_Data :: Write (char* toWhere) :

In this function, the contents of the Statistics_Data object are written to the file specified by toWhere.

3.14 void Statistics_Data :: Read (char* fromWhere):

The contents of the Statistics_Data object are read from the file specified by fromWhere in this function.

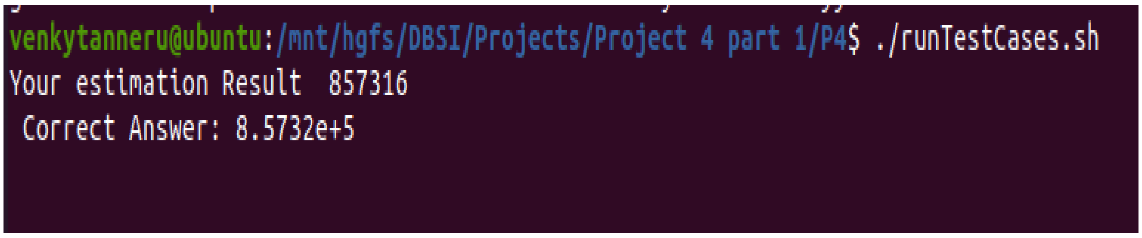
3.15 void Statistics_Data :: Apply (struct AndList* parseTree, char* relNames[], int numToJoin) :

In this function, we validate the parameters and modify the Statistics_Data object to enable the use of a single partition in a join.

4 Results:

Results are given as a **Output of Estimate** and also the output is generated in **Output.txt** file.

4.1 Output Estimate



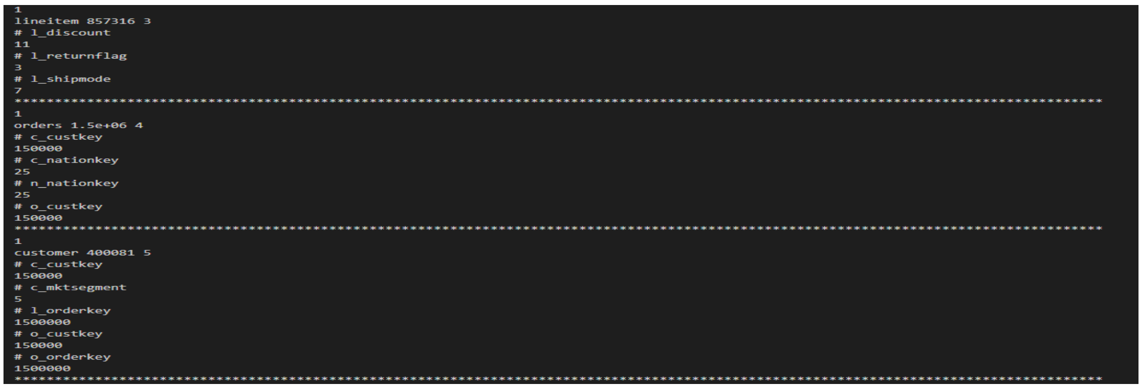
```
venkytanneru@ubuntu:/mnt/hgfs/DBSI/Projects/Project 4 part 1/P4$ ./runTestCases.sh
Your estimation Result  857316
 Correct Answer: 8.5732e+5
```

Figure 1: Output

4.2 Runtestcases file

If we run the runtestcases.sh file using the command **./runtestcases.out** then the results of all the queries were printed at **Output.txt**

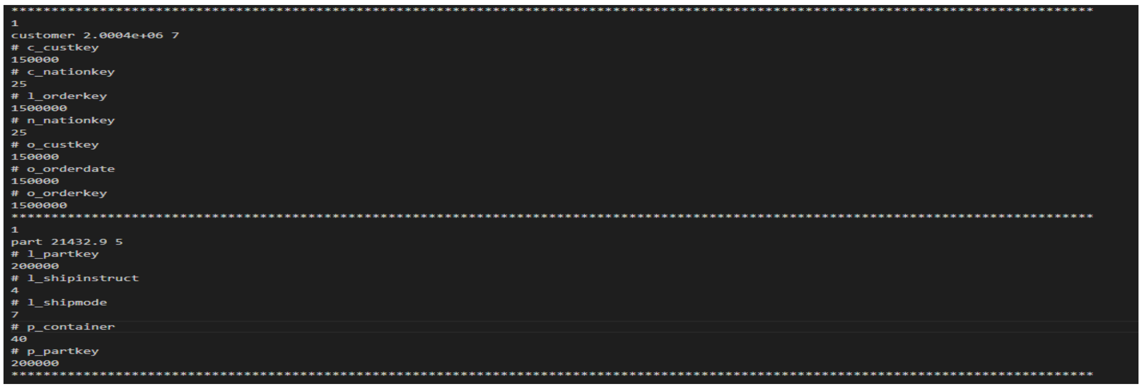
4.3 Output.txt Part 1



```
1
lineitem 857316 3
# l_discount
11
# l_returnflag
3
# l_shipmode
7
****************************************************************************************************************
1
orders 1.5e+06 4
# c_custkey
150000
# c_nationkey
25
# n_nationkey
25
# o_custkey
150000
****************************************************************************************************************
1
customer 400081 5
# c_custkey
150000
# c_mktsegment
5
# l_orderkey
1500000
# o_custkey
150000
# o_orderkey
1500000
****************************************************************************************************************
```

Figure 2: Output from Output file

4.4 Output.txt Part 2



```
****************************************************************************************************************
1
customer 2.0004e+06 7
# c_custkey
150000
# c_nationkey
25
# l_orderkey
1500000
# n_nationkey
25
# o_custkey
150000
# o_orderdate
150000
# o_orderkey
1500000
****************************************************************************************************************
1
part 21432.9 5
# l_partkey
200000
# l_shipinstruct
4
# l_shipmode
7
# p_container
40
# p_partkey
200000
****************************************************************************************************************
```

Figure 3: Output from Output file

5 Gtest:

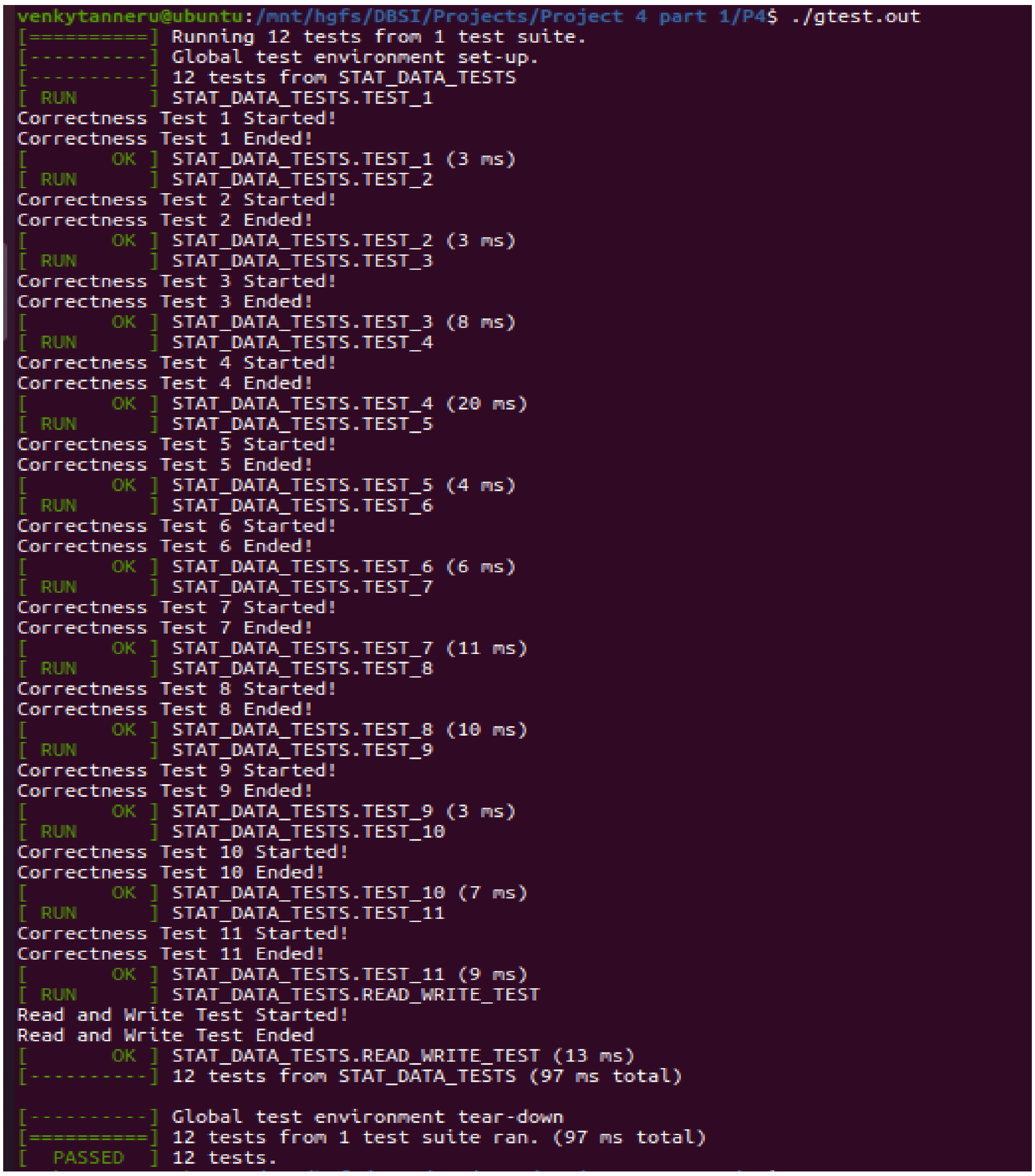
To begin running the gtest, go to the project's root directory, which includes the gtest.cpp file.

5.1 Commands To Run the Gtest Program:

- Type **make clean** and then **make gtest** to compile Google Test.
- Type **./gtest.out** to run the unit tests.
- The unit tests should run and the gtest results should be shown.

5.2 Gtest results:

we ran eleven tests on gtests. Tests include Read and Write Test, Correctness Test, Sample Test ,Estimate Tests , Add Attribute Test and with RelAdd Tests.



```
venkytanneru@ubuntu:/mnt/hgfs/DBSI/Projects/Project 4 part 1/P4$ ./gtest.out
[==========] Running 12 tests from 1 test suite.
[----------] Global test environment set-up.
[----------] 12 tests from STAT_DATA_TESTS
[ RUN      ] STAT_DATA_TESTS.TEST_1
Correctness Test 1 Started!
Correctness Test 1 Ended!
[       OK ] STAT_DATA_TESTS.TEST_1 (3 ms)
[ RUN      ] STAT_DATA_TESTS.TEST_2
Correctness Test 2 Started!
Correctness Test 2 Ended!
[       OK ] STAT_DATA_TESTS.TEST_2 (3 ms)
[ RUN      ] STAT_DATA_TESTS.TEST_3
Correctness Test 3 Started!
Correctness Test 3 Ended!
[       OK ] STAT_DATA_TESTS.TEST_3 (8 ms)
[ RUN      ] STAT_DATA_TESTS.TEST_4
Correctness Test 4 Started!
Correctness Test 4 Ended!
[       OK ] STAT_DATA_TESTS.TEST_4 (20 ms)
[ RUN      ] STAT_DATA_TESTS.TEST_5
Correctness Test 5 Started!
Correctness Test 5 Ended!
[       OK ] STAT_DATA_TESTS.TEST_5 (4 ms)
[ RUN      ] STAT_DATA_TESTS.TEST_6
Correctness Test 6 Started!
Correctness Test 6 Ended!
[       OK ] STAT_DATA_TESTS.TEST_6 (6 ms)
[ RUN      ] STAT_DATA_TESTS.TEST_7
Correctness Test 7 Started!
Correctness Test 7 Ended!
[       OK ] STAT_DATA_TESTS.TEST_7 (11 ms)
[ RUN      ] STAT_DATA_TESTS.TEST_8
Correctness Test 8 Started!
Correctness Test 8 Ended!
[       OK ] STAT_DATA_TESTS.TEST_8 (10 ms)
[ RUN      ] STAT_DATA_TESTS.TEST_9
Correctness Test 9 Started!
Correctness Test 9 Ended!
[       OK ] STAT_DATA_TESTS.TEST_9 (3 ms)
[ RUN      ] STAT_DATA_TESTS.TEST_10
Correctness Test 10 Started!
Correctness Test 10 Ended!
[       OK ] STAT_DATA_TESTS.TEST_10 (7 ms)
[ RUN      ] STAT_DATA_TESTS.TEST_11
Correctness Test 11 Started!
Correctness Test 11 Ended!
[       OK ] STAT_DATA_TESTS.TEST_11 (9 ms)
[ RUN      ] STAT_DATA_TESTS.READ_WRITE_TEST
Read and Write Test Started!
Read and Write Test Ended
[       OK ] STAT_DATA_TESTS.READ_WRITE_TEST (13 ms)
[----------] 12 tests from STAT_DATA_TESTS (97 ms total)

[----------] Global test environment tear-down
[==========] 12 tests from 1 test suite ran. (97 ms total)
[  PASSED  ] 12 tests.
```

Figure 4: Gtest Results