COP 6726 – Database System Implementation
Project 3: Relational Operations

Group Members:

Sanjay Reddy Banda, UF ID: 5878-2239

Suprith Reddy Gurudu, UF ID: 9961-2134

Compilation and Execution:

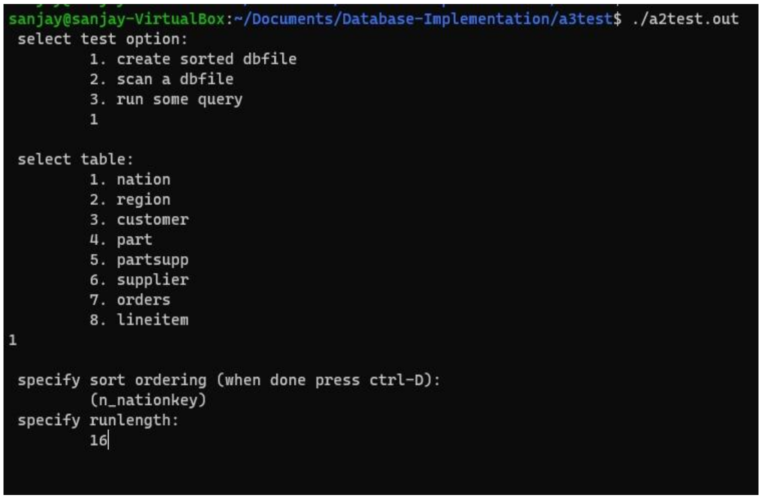
Bin files are generated by a2-test.cc, run the following command:

Compile:

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

Run:

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



```
sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a3test$ ./a2test.out
 select test option:
        1. create sorted dbfile
        2. scan a dbfile
        3. run some query
        1

 select table:
        1. nation
        2. region
        3. customer
        4. part
        5. partsupp
        6. supplier
        7. orders
        8. lineitem
1

 specify sort ordering (when done press ctrl-D):
        (n_nationkey)
 specify runlength:
        16
```

To compile the code, run the following command:

```
>> make
```

To execute the test.cc code, change the directory to the specific folder (a3 test) and run the following command:

```
>> ./test.out <query(1-5)>
```

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: RelOp.cc

1. *void *performOperation(void *obj) method –*
   It starts the specific relational operator function where relational operation is chosen by the type of 'obj' during runtime.

2. *void RelationalOp::WaitUntilDone () method –*
   This function keeps the current thread waiting until the specified relational operation thread is finished its execution.

3. *void SelectFile::Run (DBFile &inFile, Pipe &outPipe, CNF &selOp, Record &literal) method –*
   This function initializes all the variables needed to perform select file operation and spawns the thread which performs select file operation.

4. *void SelectFile::Start() method –*
   This function gets all the records from the specified DB file and inserts the records into an out pipe which satisfies the given CNF.

5. *void SelectPipe::Run (Pipe &inPipe, Pipe &outPipe, CNF &selOp, Record &literal) method –*
   This function initializes all the variables needed to perform select pipe operation and spawns the thread which performs select pipe operation.

6. *void SelectPipe::Start() method –*
   This function gets all the records from the inpipe and inserts the records into an out pipe which satisfies the given CNF.

7. *void Project :: Run (Pipe &inPipe, Pipe &outPipe, int *keepMe, int numAttsInput, int numAttsOutput) method –*
   This function initializes all the variables needed to perform project relational operation and spawns the thread which performs project relational operation.

8. *void Project :: Start() method –*
   This function gets all the records from the inpipe and inserts the records into an out pipe with specified attributes (subset of actual attributes of a record).

9. *void Join :: Run (Pipe &inPipeL, Pipe &inPipeR, Pipe &outPipe, CNF &selOp, Record &literal) method –*
   This function initializes all the variables needed to perform join relational operation and spawns the thread which performs join relational operation.

10. *void Join :: Start() method –*

This function joins all the records from the inpipe left and inpipe right which satisfies the given CNF and inserts the resultant records into the out pipe.

11. *void DuplicateRemoval :: Run (Pipe &inPipe, Pipe &outPipe, Schema &mySchema) method –*
This function initializes all the variables needed to perform distinct relational operation and spawns the thread which performs distinct relational operation.

12. *void DuplicateRemoval :: Start() method –*
This function gets all the records from the inpipe and returns the distinct records into an out pipe.

13. *void Sum :: Run (Pipe &inPipe, Pipe &outPipe, Function &computeMe)method –*
This function initializes all the variables needed to perform sum relational operation and spawns the thread which performs sum relational operation.

14. *void Sum :: Start() method –*
This function gets all the records from the inpipe and returns the sum of the given attributes as a single record into an out pipe.

15. *void GroupBy :: Run (Pipe &inPipe, Pipe &outPipe, OrderMaker &groupAtts, Function &computeMe) method –*
This function initializes all the variables needed to perform groupby relational operation and spawns the thread which performs groupby relational operation.

16. *void GroupBy :: Start() method –*
This function gets all the records from the inpipe and returns the groupby of the given attributes as a result into an out pipe.

17. *void WriteOut :: Run (Pipe &inPipe, FILE *outFile, Schema &mySchema) method –*
This function initializes all the variables needed to perform write to file operation and spawns the thread which performs write to file operation.

18. *void WriteOut :: Start() method –*
This function gets all the records from the inpipe and writes all the records to the specified file with the given schema.

Filename: gtests.cc

TEST(RELATIONALOPERATIONS,DISTINCT) –
Google test for validating the distinct relational operator function scenario. It verifies by total number of records returned by the function.

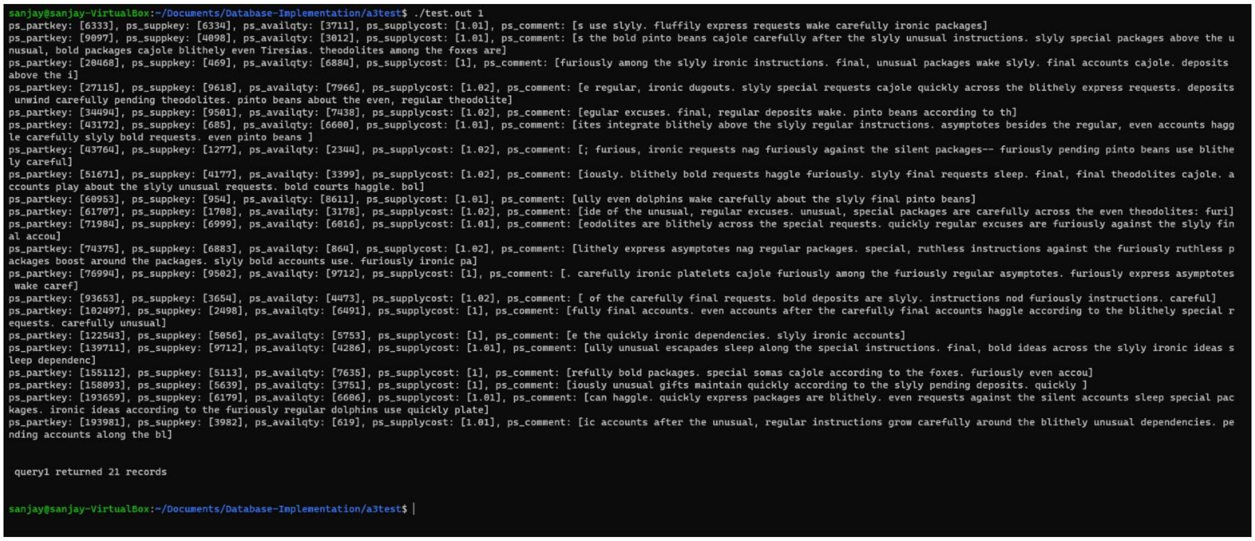
```
TEST(RELATIONALOPERATIONS,SELECT) –
      Google test for validating the select relational operator
function scenario. It verifies by total number of records returned by
the function.
TEST(RELATIONALOPERATIONS,PROJECT) –
      Google test for validating the project relational operator
function scenario. It verifies by total number of records returned by
the function.
TEST(RELATIONALOPERATIONS,SUM) –
      Google test for validating the sum relational operator function
scenario. It verifies by total number of records returned by the
function.
```

Results for the Test Cases:

Test Case 1:

Input - select * from partsupp where ps_supplycost <1.03

Note: It was mentioned in the test.cc that query returns 31 records, but we are getting 21 records. We have verified in a different manner, where we got 21 records as the result. So, the anomaly might be due to the data in the tables.

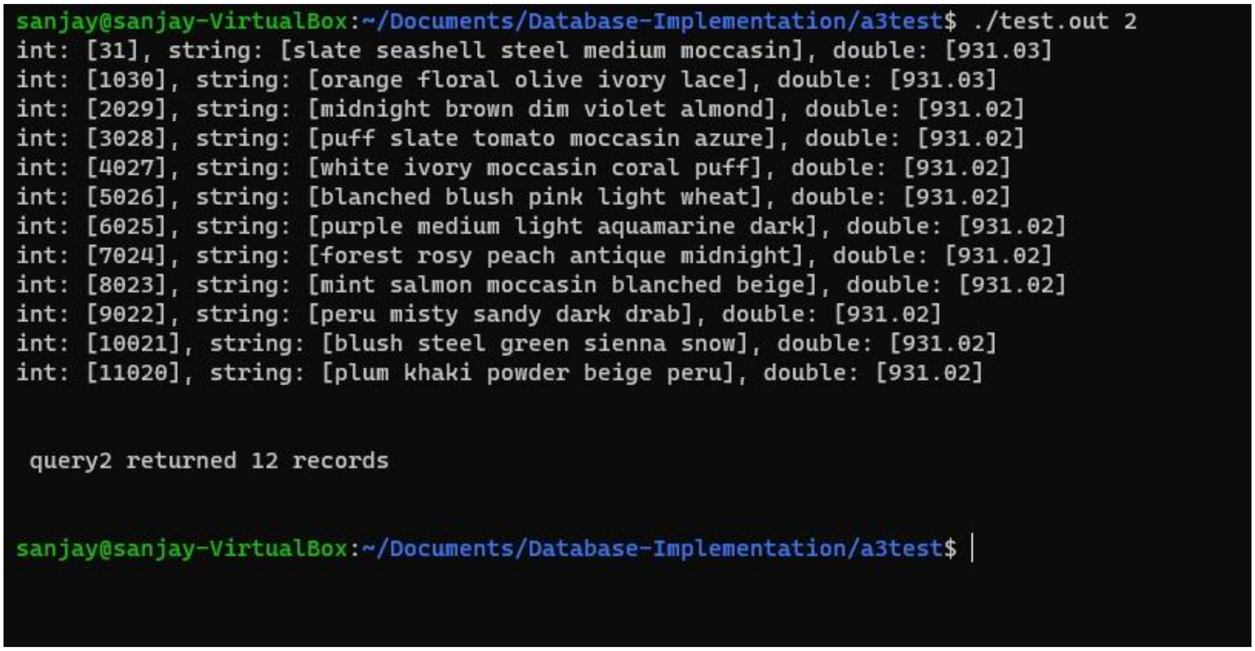


```
sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a3test$ ./test.out 1
ps_partkey: [6333], ps_suppkey: [6334], ps_availqty: [3711], ps_supplycost: [1.01], ps_comment: [s use slyly. fluffily express requests wake carefully ironic packages]
ps_partkey: [9097], ps_suppkey: [4098], ps_availqty: [3012], ps_supplycost: [1.01], ps_comment: [s the bold pinto beans cajole carefully after the slyly unusual instructions. slyly special packages above the u
nusual, bold packages cajole blithely even Tiresias. theodolites among the foxes are]
ps_partkey: [20468], ps_suppkey: [469], ps_availqty: [6884], ps_supplycost: [1], ps_comment: [furiously among the slyly ironic instructions. final, unusual packages wake slyly. final accounts cajole. deposits
above the i]
ps_partkey: [27115], ps_suppkey: [9618], ps_availqty: [7966], ps_supplycost: [1.02], ps_comment: [e regular, ironic dugouts. slyly special requests cajole quickly across the blithely express requests. deposits
 unwind carefully pending theodolites. pinto beans about the even, regular theodolite]
ps_partkey: [34494], ps_suppkey: [9501], ps_availqty: [7438], ps_supplycost: [1.02], ps_comment: [egular excuses. final, regular deposits wake. pinto beans according to th]
ps_partkey: [43172], ps_suppkey: [685], ps_availqty: [6600], ps_supplycost: [1.01], ps_comment: [ites integrate blithely above the slyly regular instructions. asymptotes besides the regular, even accounts hagg
le carefully slyly bold requests. even pinto beans ]
ps_partkey: [43764], ps_suppkey: [1277], ps_availqty: [2344], ps_supplycost: [1.02], ps_comment: [; furious, ironic requests nag furiously against the silent packages-- furiously pending pinto beans use blithe
ly careful]
ps_partkey: [51671], ps_suppkey: [4177], ps_availqty: [3399], ps_supplycost: [1.02], ps_comment: [iously. blithely bold requests haggle furiously. slyly final requests sleep. final, final theodolites cajole. a
ccounts play about the slyly unusual requests. bold courts haggle. bol]
ps_partkey: [60953], ps_suppkey: [954], ps_availqty: [8611], ps_supplycost: [1.01], ps_comment: [ully even dolphins wake carefully about the slyly final pinto beans]
ps_partkey: [61707], ps_suppkey: [1708], ps_availqty: [3178], ps_supplycost: [1.02], ps_comment: [ide of the unusual, regular excuses. unusual, special packages are carefully across the even theodolites: furi]
ps_partkey: [71984], ps_suppkey: [6999], ps_availqty: [6016], ps_supplycost: [1.01], ps_comment: [eodolites are blithely across the special requests. quickly regular excuses are furiously against the slyly fin
al accou]
ps_partkey: [74375], ps_suppkey: [6883], ps_availqty: [864], ps_supplycost: [1.02], ps_comment: [lithely express asymptotes nag regular packages. special, ruthless instructions against the furiously ruthless p
ackages boost around the packages. slyly bold accounts use. furiously ironic pa]
ps_partkey: [76994], ps_suppkey: [9502], ps_availqty: [9712], ps_supplycost: [1], ps_comment: [. carefully ironic platelets cajole furiously among the furiously regular asymptotes. furiously express asymptotes
 wake caref]
ps_partkey: [93653], ps_suppkey: [3654], ps_availqty: [4473], ps_supplycost: [1.02], ps_comment: [ of the carefully final requests. bold deposits are slyly. instructions nod furiously instructions. careful]
ps_partkey: [102497], ps_suppkey: [2498], ps_availqty: [6491], ps_supplycost: [1], ps_comment: [fully final accounts. even accounts after the carefully final accounts haggle according to the blithely special r
equests. carefully unusual]
ps_partkey: [122543], ps_suppkey: [5056], ps_availqty: [5753], ps_supplycost: [1], ps_comment: [e the quickly ironic dependencies. slyly ironic accounts]
ps_partkey: [139711], ps_suppkey: [9712], ps_availqty: [4286], ps_supplycost: [1.01], ps_comment: [ully unusual escapades sleep along the special instructions. final, bold ideas across the slyly ironic ideas s
leep dependenc]
ps_partkey: [155112], ps_suppkey: [5113], ps_availqty: [7635], ps_supplycost: [1], ps_comment: [refully bold packages. special somas cajole according to the foxes. furiously even accou]
ps_partkey: [158093], ps_suppkey: [5639], ps_availqty: [3751], ps_supplycost: [1], ps_comment: [iously unusual gifts maintain quickly according to the slyly pending deposits. quickly ]
ps_partkey: [193659], ps_suppkey: [6179], ps_availqty: [6606], ps_supplycost: [1.01], ps_comment: [can haggle. quickly express packages are blithely. even requests against the silent accounts sleep special pac
kages. ironic ideas according to the furiously regular dolphins use quickly plate]
ps_partkey: [193981], ps_suppkey: [3982], ps_availqty: [619], ps_supplycost: [1.01], ps_comment: [ic accounts after the unusual, regular instructions grow carefully around the blithely unusual dependencies. pe
nding accounts along the bl]


 query1 returned 21 records


sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a3test$ |
```

Test Case 2:

Input - select p_partkey(0), p_name(1), p_retailprice(7) from part where (p_retailprice > 931.01) AND (p_retailprice < 931.3);

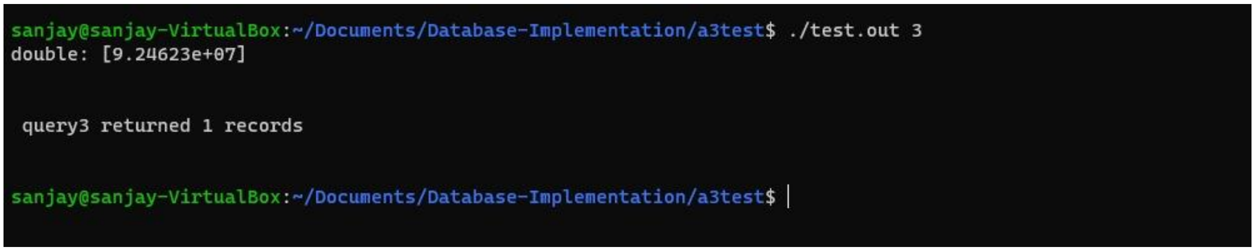
Note: It was mentioned in the test.cc that query returns 22 records, but we are getting 12 records. We have verified in a different manner, where we got 12 records as the result. So, the anomaly might be due to the data in the tables.



```
sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a3test$ ./test.out 2
int: [31], string: [slate seashell steel medium moccasin], double: [931.03]
int: [1030], string: [orange floral olive ivory lace], double: [931.03]
int: [2029], string: [midnight brown dim violet almond], double: [931.02]
int: [3028], string: [puff slate tomato moccasin azure], double: [931.02]
int: [4027], string: [white ivory moccasin coral puff], double: [931.02]
int: [5026], string: [blanched blush pink light wheat], double: [931.02]
int: [6025], string: [purple medium light aquamarine dark], double: [931.02]
int: [7024], string: [forest rosy peach antique midnight], double: [931.02]
int: [8023], string: [mint salmon moccasin blanched beige], double: [931.02]
int: [9022], string: [peru misty sandy dark drab], double: [931.02]
int: [10021], string: [blush steel green sienna snow], double: [931.02]
int: [11020], string: [plum khaki powder beige peru], double: [931.02]


 query2 returned 12 records


sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a3test$ |
```

Test Case 3:

Input - select sum (s_acctbal + (s_acctbal * 1.05)) from supplier;

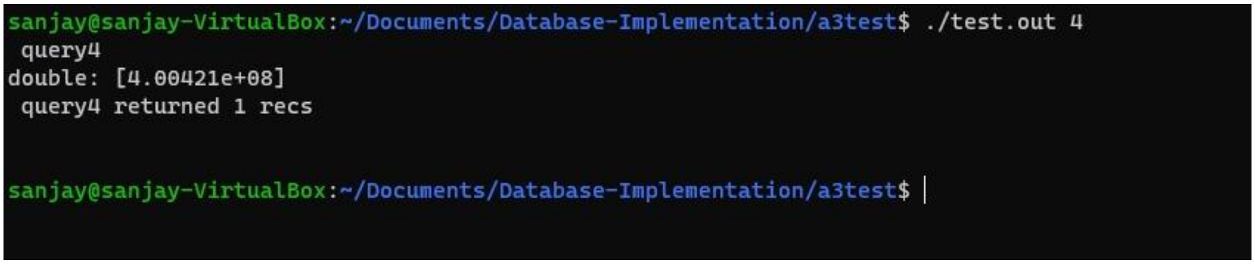


```
sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a3test$ ./test.out 3
double: [9.24623e+07]


 query3 returned 1 records


sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a3test$ |
```

Test Case 4:

Input - select sum (ps_supplycost) from supplier, partsupp where s_suppkey = ps_suppkey;

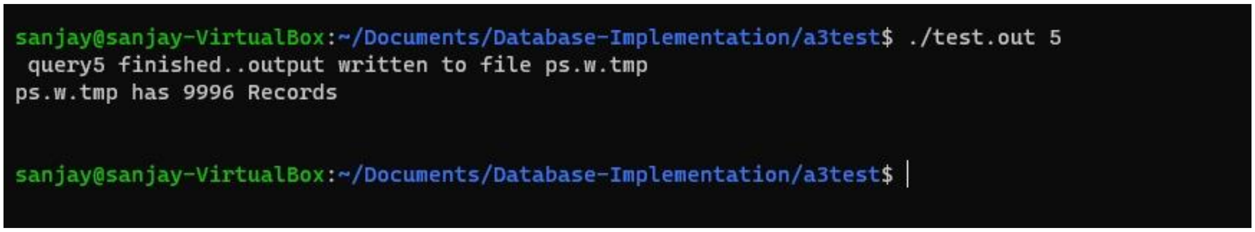
Note: It was mentioned in the test.cc that query returns 4.00406e+08 as sum, but we are getting 4.00421e+08 as sum. We have verified in a different manner, where we got 4.00421e+08 as the result. So, the anomaly might be due to the data in the tables.



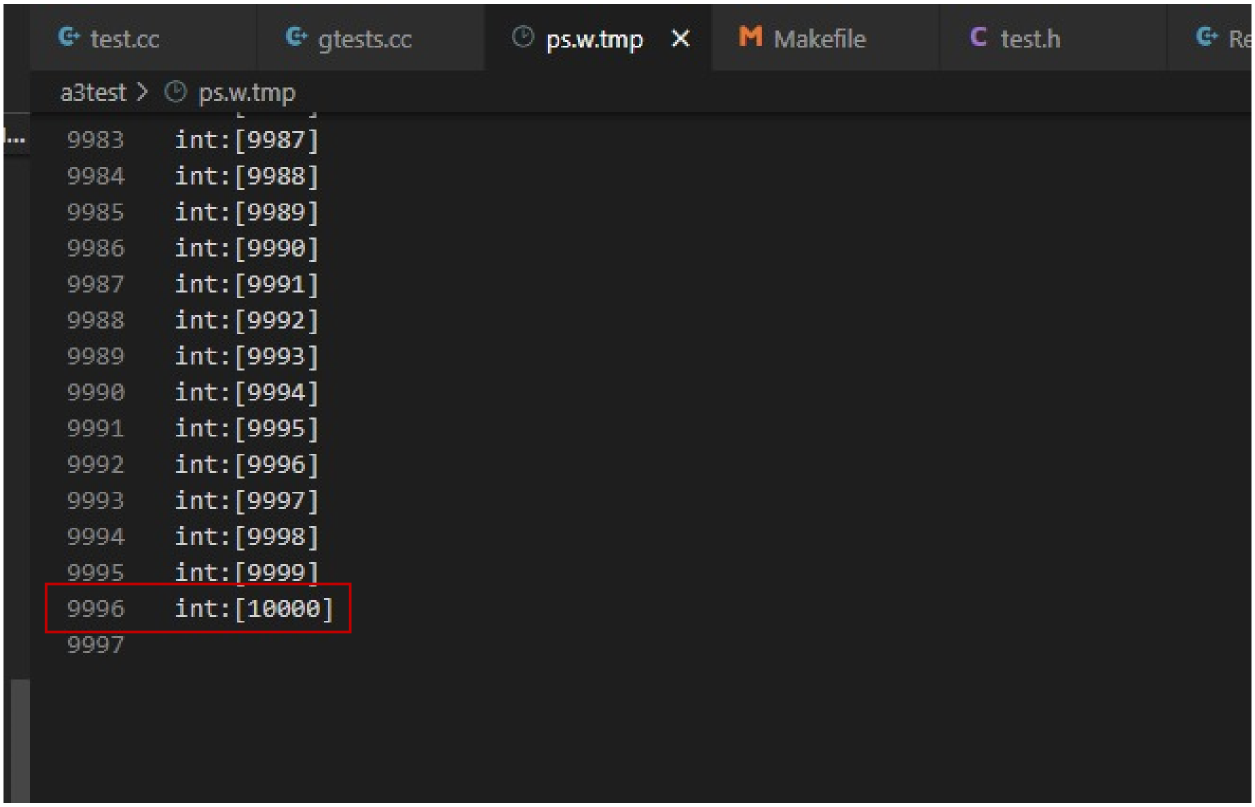
```
sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a3test$ ./test.out 4
 query4
double: [4.00421e+08]
 query4 returned 1 recs


sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a3test$ |
```

Test Case 5:

Input - select distinct ps_suppkey from partsupp where ps_supplycost < 100.11;

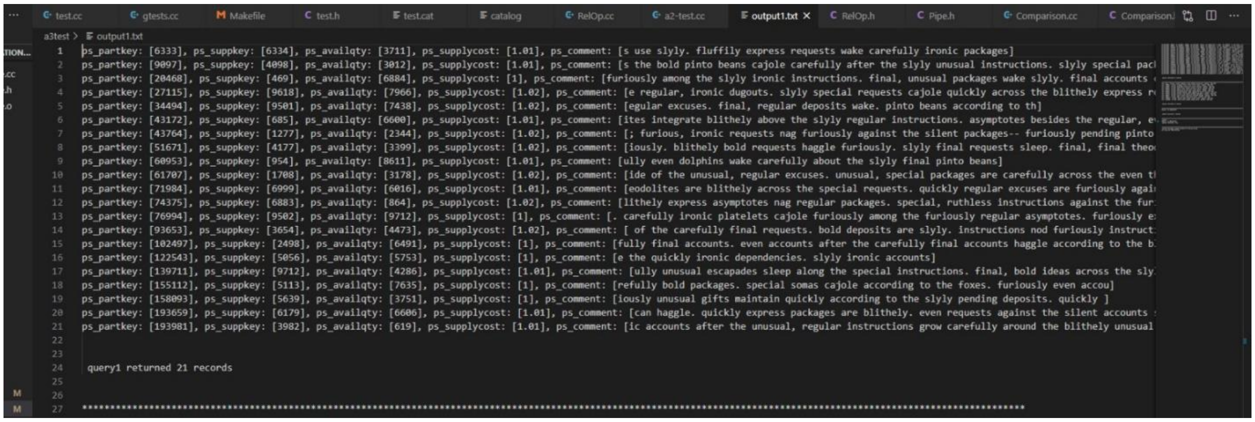


```
sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a3test$ ./test.out 5
 query5 finished..output written to file ps.w.tmp
ps.w.tmp has 9996 Records

sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a3test$ 
```

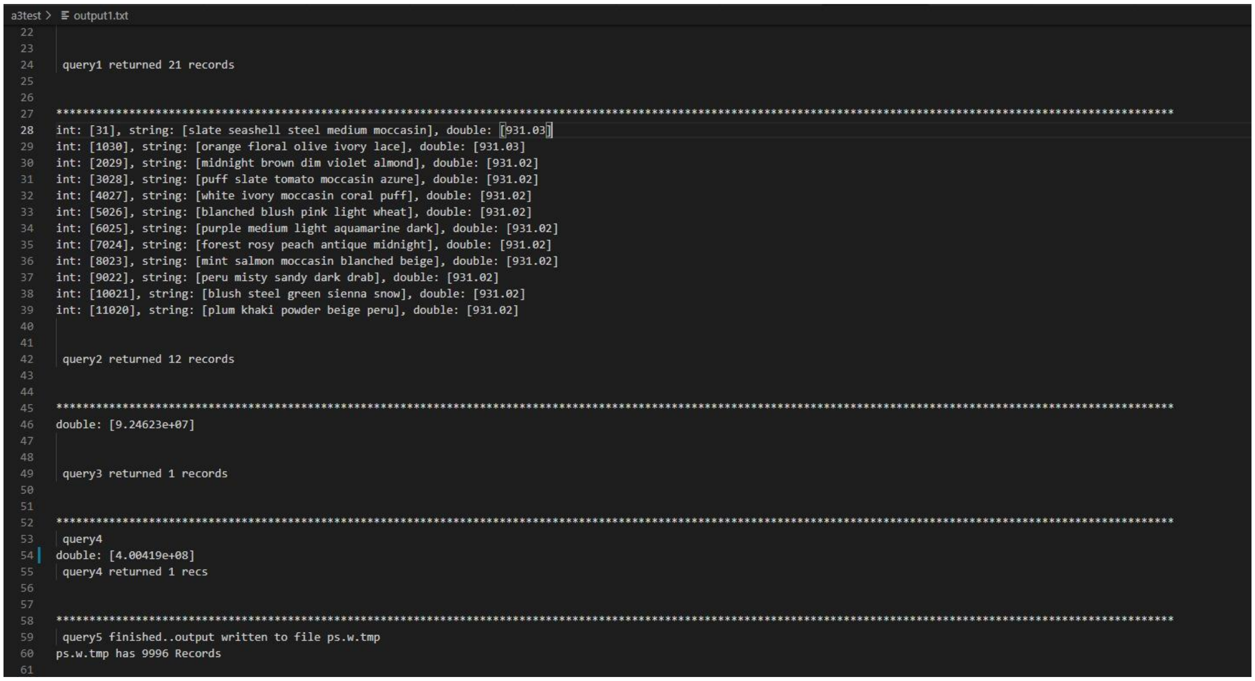


```
9983    int:[9987]
9984    int:[9988]
9985    int:[9989]
9986    int:[9990]
9987    int:[9991]
9988    int:[9992]
9989    int:[9993]
9990    int:[9994]
9991    int:[9995]
9992    int:[9996]
9993    int:[9997]
9994    int:[9998]
9995    int:[9999]
9996    int:[10000]
9997
```

Results for output1.txt:

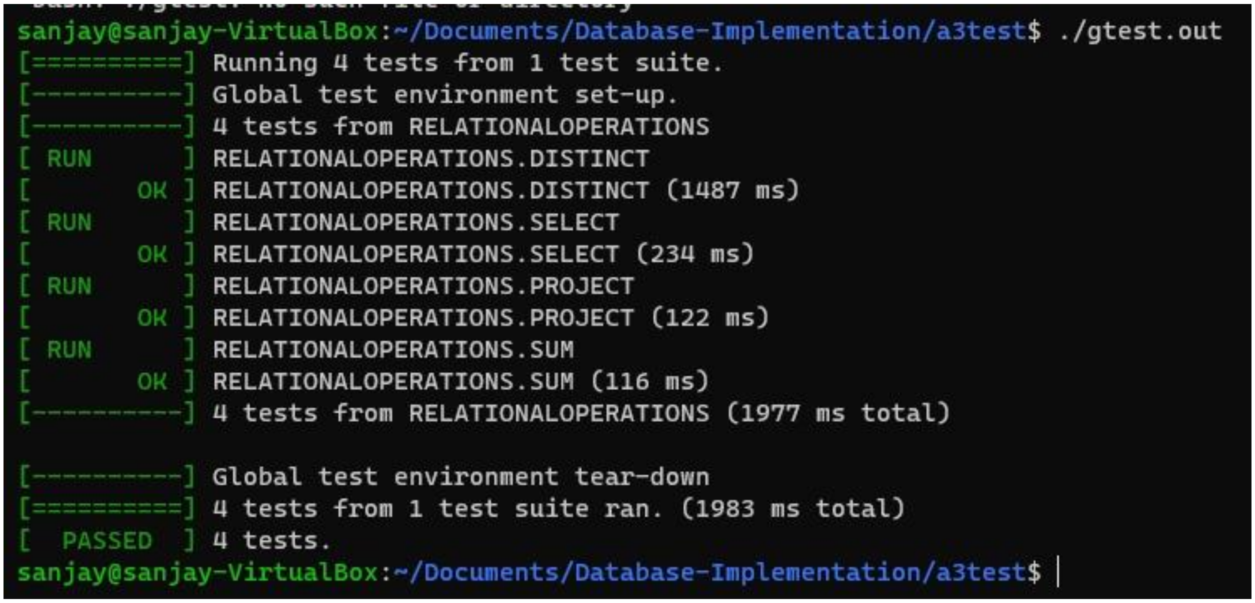


```
ps_partkey: [6333], ps_suppkey: [6334], ps_availqty: [3711], ps_supplycost: [1.01], ps_comment: [s use slyly. fluffily express requests wake carefully ironic packages]
ps_partkey: [9097], ps_suppkey: [4098], ps_availqty: [3012], ps_supplycost: [1.01], ps_comment: [s the bold pinto beans cajole carefully after the slyly unusual instructions. slyly special pack
ps_partkey: [20468], ps_suppkey: [469], ps_availqty: [6884], ps_supplycost: [1], ps_comment: [furiously among the slyly ironic instructions. final, unusual packages wake slyly. final accounts
ps_partkey: [27115], ps_suppkey: [9618], ps_availqty: [7966], ps_supplycost: [1.02], ps_comment: [e regular, ironic dugouts. slyly special requests cajole quickly across the blithely express r
ps_partkey: [34494], ps_suppkey: [9501], ps_availqty: [7438], ps_supplycost: [1.02], ps_comment: [egular excuses. final, regular deposits wake. pinto beans according to th]
ps_partkey: [43172], ps_suppkey: [685], ps_availqty: [6600], ps_supplycost: [1.01], ps_comment: [ites integrate blithely above the slyly regular instructions. asymptotes besides the regular, e
ps_partkey: [43764], ps_suppkey: [1277], ps_availqty: [2344], ps_supplycost: [1.02], ps_comment: [; furious, ironic requests nag furiously against the silent packages-- furiously pending pinto
ps_partkey: [51671], ps_suppkey: [4177], ps_availqty: [3399], ps_supplycost: [1.02], ps_comment: [iously. blithely bold requests haggle furiously. slyly final requests sleep. final, final theo
ps_partkey: [60953], ps_suppkey: [954], ps_availqty: [8611], ps_supplycost: [1.01], ps_comment: [ully even dolphins wake carefully about the slyly final pinto beans]
ps_partkey: [61707], ps_suppkey: [1708], ps_availqty: [3178], ps_supplycost: [1.02], ps_comment: [ide of the unusual, regular excuses. unusual, special packages are carefully across the even t
ps_partkey: [71984], ps_suppkey: [6999], ps_availqty: [6016], ps_supplycost: [1.01], ps_comment: [eodolites are blithely across the special requests. quickly regular excuses are furiously agai
ps_partkey: [74375], ps_suppkey: [6883], ps_availqty: [864], ps_supplycost: [1.02], ps_comment: [lithely express asymptotes nag regular packages. special, ruthless instructions against the fur
ps_partkey: [76994], ps_suppkey: [9502], ps_availqty: [9712], ps_supplycost: [1], ps_comment: [. carefully ironic platelets cajole furiously among the furiously regular asymptotes. furiously e
ps_partkey: [93653], ps_suppkey: [3654], ps_availqty: [4473], ps_supplycost: [1.02], ps_comment: [ of the carefully final requests. bold deposits are slyly. instructions nod furiously instruct
ps_partkey: [102497], ps_suppkey: [2498], ps_availqty: [6491], ps_supplycost: [1], ps_comment: [fully final accounts. even accounts after the carefully final accounts haggle according to the b
ps_partkey: [122543], ps_suppkey: [5056], ps_availqty: [5753], ps_supplycost: [1], ps_comment: [e the quickly ironic dependencies. slyly ironic accounts]
ps_partkey: [139711], ps_suppkey: [9712], ps_availqty: [4286], ps_supplycost: [1.01], ps_comment: [ully unusual escapades sleep along the special instructions. final, bold ideas across the sly
ps_partkey: [155112], ps_suppkey: [5113], ps_availqty: [7635], ps_supplycost: [1], ps_comment: [refully bold packages. special somas cajole according to the foxes. furiously even accou]
ps_partkey: [158093], ps_suppkey: [5639], ps_availqty: [3751], ps_supplycost: [1], ps_comment: [iously unusual gifts maintain quickly according to the slyly pending deposits. quickly ]
ps_partkey: [193659], ps_suppkey: [6179], ps_availqty: [6606], ps_supplycost: [1.01], ps_comment: [can haggle. quickly express packages are blithely. even requests against the silent accounts
ps_partkey: [193981], ps_suppkey: [3982], ps_availqty: [619], ps_supplycost: [1.01], ps_comment: [ic accounts after the unusual, regular instructions grow carefully around the blithely unusual



 query1 returned 21 records


************************************************************************************************************************************************************************
```



```


 query1 returned 21 records


************************************************************************************************************************************************************************
int: [31], string: [slate seashell steel medium moccasin], double: [931.03]
int: [1030], string: [orange floral olive ivory lace], double: [931.03]
int: [2029], string: [midnight brown dim violet almond], double: [931.02]
int: [3028], string: [puff slate tomato moccasin azure], double: [931.02]
int: [4027], string: [white ivory moccasin coral puff], double: [931.02]
int: [5026], string: [blanched blush pink light wheat], double: [931.02]
int: [6025], string: [purple medium light aquamarine dark], double: [931.02]
int: [7024], string: [forest rosy peach antique midnight], double: [931.02]
int: [8023], string: [mint salmon moccasin blanched beige], double: [931.02]
int: [9022], string: [peru misty sandy dark drab], double: [931.02]
int: [10021], string: [blush steel green sienna snow], double: [931.02]
int: [11020], string: [plum khaki powder beige peru], double: [931.02]


 query2 returned 12 records


************************************************************************************************************************************************************************
double: [9.24623e+07]


 query3 returned 1 records


************************************************************************************************************************************************************************
 query4
double: [4.00419e+08]
 query4 returned 1 recs


************************************************************************************************************************************************************************
 query5 finished..output written to file ps.w.tmp
ps.w.tmp has 9996 Records

```

Results for gTests:



```
bash: ./gtest: No such file or directory
sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a3test$ ./gtest.out
[==========] Running 4 tests from 1 test suite.
[----------] Global test environment set-up.
[----------] 4 tests from RELATIONALOPERATIONS
[ RUN      ] RELATIONALOPERATIONS.DISTINCT
[       OK ] RELATIONALOPERATIONS.DISTINCT (1487 ms)
[ RUN      ] RELATIONALOPERATIONS.SELECT
[       OK ] RELATIONALOPERATIONS.SELECT (234 ms)
[ RUN      ] RELATIONALOPERATIONS.PROJECT
[       OK ] RELATIONALOPERATIONS.PROJECT (122 ms)
[ RUN      ] RELATIONALOPERATIONS.SUM
[       OK ] RELATIONALOPERATIONS.SUM (116 ms)
[----------] 4 tests from RELATIONALOPERATIONS (1977 ms total)

[----------] Global test environment tear-down
[==========] 4 tests from 1 test suite ran. (1983 ms total)
[  PASSED  ] 4 tests.
sanjay@sanjay-VirtualBox:~/Documents/Database-Implementation/a3test$
```