Data Base System Implementation
COP-6726

Submitted by:
Naman Arora
UFID: 3979-0439
Nikunj Sarda
UFID: 9360-6581

Introduction:

This document is intended to present the outcomes and the process of the completed work for the Project 3. The relational operations for DB file implementation have been successfully materialized and tested. The repository for all the work is present here.

Source File Hierarchy:

The directories are arranged in a flat manner with exception of a tpch-dbgen project directory. The tpch-dbgen is the git repository for the TPCH sub program. Please make sure the *.tbl files are present BEFORE test is run.

Some Points to Note:

→ Note 1: flex version equal or higher than 2.6 is needed to run the above implementation and not get below error during compilation phase:

“lexer_func.l:67:1: error: ‘yyfunclineno’ undeclared (first use in this function); did you mean ‘yyfuncleng’?”

while running the make to generate test.out file. If such an error arises, however, please uncomment line 16 in ‘lexer_func.l’.

→ Note 2: Please use “add a lot” option, set as default, in sorted_file_test.out since some tables, notably the partsupp.tbl get killed by the OS due to large number of processes.

The compilation process:

The compilation process is accomplished by a recursive call to Makefiles in each subdirectory by the Makefile in the src/ directory.

The provided NamanArora_NikunjSarda_p3.zip has one sub-directory, src/.

```
$> unzip NamanArora_NikunjSarda_p3.zip
$> cd proj3
```

Now to build:

```
$> make
```

The above command will create all the binaries, test.out, gtest.out, test_heap.out and test_bigq.out along with tpch-dbgen/dbgen.

To generate tpch-dbgen files (tpch-dbgen/ sub-dir is always compiled with all of the above targets)

```
$> cd tpch-dbgen/
$> ./dbgen <options> -s <size>
$> mv *.tbl ..
$> sync
```

To generate heap database files required for running the test driver,

```
$> make heap_gen.out
$> ./create_heap_file.sh #  All the heap database files will be available
```

To generate binary for creating sorted database files:

```
$> make sorted_test_file.out # This creates the binary driver for project 2 milestone 2
```

To run the associated Gtests for the current submission:

```
$> make gtest.out
$> ./gtest.out
```

To clean the repository, excluding *.bin and *.tbl files,

```
$> make clean
```

To clean the repository, including *.bin and *.tbl files

```
$> make distclean
```

Change log:

- Added group_by.h and group_by.cc as the implementation of the GroupBy operation
- Added join.h and join.cc as the implementation of the Join operation
- Added rem_dup.h and rem_dup.cc as the implementation of the remove Duplicate operation
- Added sum.h and sum.cc as the implementation of the sum operation
- Added sel_file.h and sel_file.cc as the implementation to select file
- Added sel_pipe.h and sel_pipe.cc as the implementation to select pipe
- Added write_out.h and write_out.cc as the implementation of the writing a file
- Added project.h and project.cc as the implementation to select project
- Added an overloaded constructor for class OrderMaker which generates instances that help in sorting based on a single attribute.
- Added a destructor for BigQ that waits for the worker thread to join back.
- Added a Record :: deserialize function which returns the record in form of a character array.

group_by.cc:

- void mk_grps(struct grp_args *arg, vector<Record *> &grps, vector<struct grp_args *> &args):This functions receives vectors in form of pointers pointing to constant memory location so that vectors are not copies across functions
- void push_results(struct grp_args *arg, vector<struct grp_args *> &args): Put the results in the out pipe
- void *grp_thr(void *a): Function to create a thread for the group
- GroupBy :: GroupBy(): Constructor of the class to initialize initial argument
- GroupBy :: ~GroupBy(): Destructor of the class to destroy class variable
- void GroupBy :: Run(Pipe *in_pipe, Pipe *out_pipe, OrderMaker *order, Function *func): Public function called from outside the class to start the GroupBy operation
- void GroupBy :: WaitUntilDone(): Function to pause future threads till current thread is completed

- void GroupBy :: Use_n_Pages(int n): Function to decide number of pages to be used

join.cc:

- void sort_merge(struct join_args *arg): Function to merge 2 given records in sorted manner
- void nested_loop(struct join_args *arg): Function to do record join using nested loop
- void *join_thr(void *a): Function to create a thread for the Join
- Join :: Join(): Constructor of the class to initialize initial argument
- Join :: ~Join(): Destructor of the class to destroy class variable
- void Join :: Run(Pipe *in_pipeL, Pipe *in_pipeR, Pipe *out_pipe, CNF *cnf, Record *literal, Schema *schL, Schema *schR): Public function called from outside the class to start the Join operation
- void Join :: WaitUntilDone(): Function to pause future threads till current thread is completed
- void Join :: Use_n_Pages(int n): Function to decide number of pages to be used

project.cc:

- void *project_thr(void *a): Function to create a thread for the Project
- Project :: Project(): Constructor of the class to initialize initial argument
- Project :: ~Project(): Destructor of the class to destroy class variable
- void Project :: Run(Pipe *in_pipe, Pipe *out_pipe, int *keep, int atts_in, int atts_out): Public function called from outside the class to start the Project selection
- void Project :: WaitUntilDone(): Function to pause future threads till current thread is completed
- void Project :: Use_n_Pages(int n): Function to decide number of pages to be used

rem_dup.cc:

- void *dup_thr(void *a): Function to create a thread for duplicate removal
- DuplicateRemoval :: DuplicateRemoval(): Constructor of the class to initialize initial argument
- DuplicateRemoval :: ~DuplicateRemoval(): Destructor of the class to destroy class variable
- void DuplicateRemoval :: Run(Pipe *in_pipe, Pipe *out_pipe, Schema *sch): Public function called from outside the class to start the duplicate removal operation
- void DuplicateRemoval :: WaitUntilDone(): Function to pause future threads till current thread is completed
- void DuplicateRemoval :: Use_n_Pages(int n): Function to decide number of pages to be used

sel_file.cc:

- void *file_thr(void *input): Function to create a thread for file selection
- SelectFile :: SelectFile(): Constructor of the class to initialize initial argument
- SelectFile :: ~SelectFile(): Destructor of the class to destroy class variable
- void SelectFile :: Run(DBFile *dbf, Pipe *out_pipe, CNF *cnf, Record *literal): Public function called from outside the class to start the File selection
- void SelectFile :: WaitUntilDone(): Function to pause future threads till current thread is completed

- void SelectFile :: Use_n_Pages(int n): Function to decide number of pages to be used

sel_pipe.cc:

- void *pipe_thr(void *a): Function to create a thread for pipe selection
- SelectPipe :: SelectPipe(): Constructor of the class to initialize initial argument
- SelectPipe :: ~SelectPipe(): Destructor of the class to destroy class variable
- void SelectPipe :: Run(Pipe *in_pipe, Pipe *out_pipe, CNF *cnf, Record *literal): Public function called from outside the class to start the Pipe selection
- void SelectPipe :: WaitUntilDone(): Function to pause future threads till current thread is completed
- void SelectPipe :: Use_n_Pages(int n): Function to decide number of pages to be used

sum.cc:

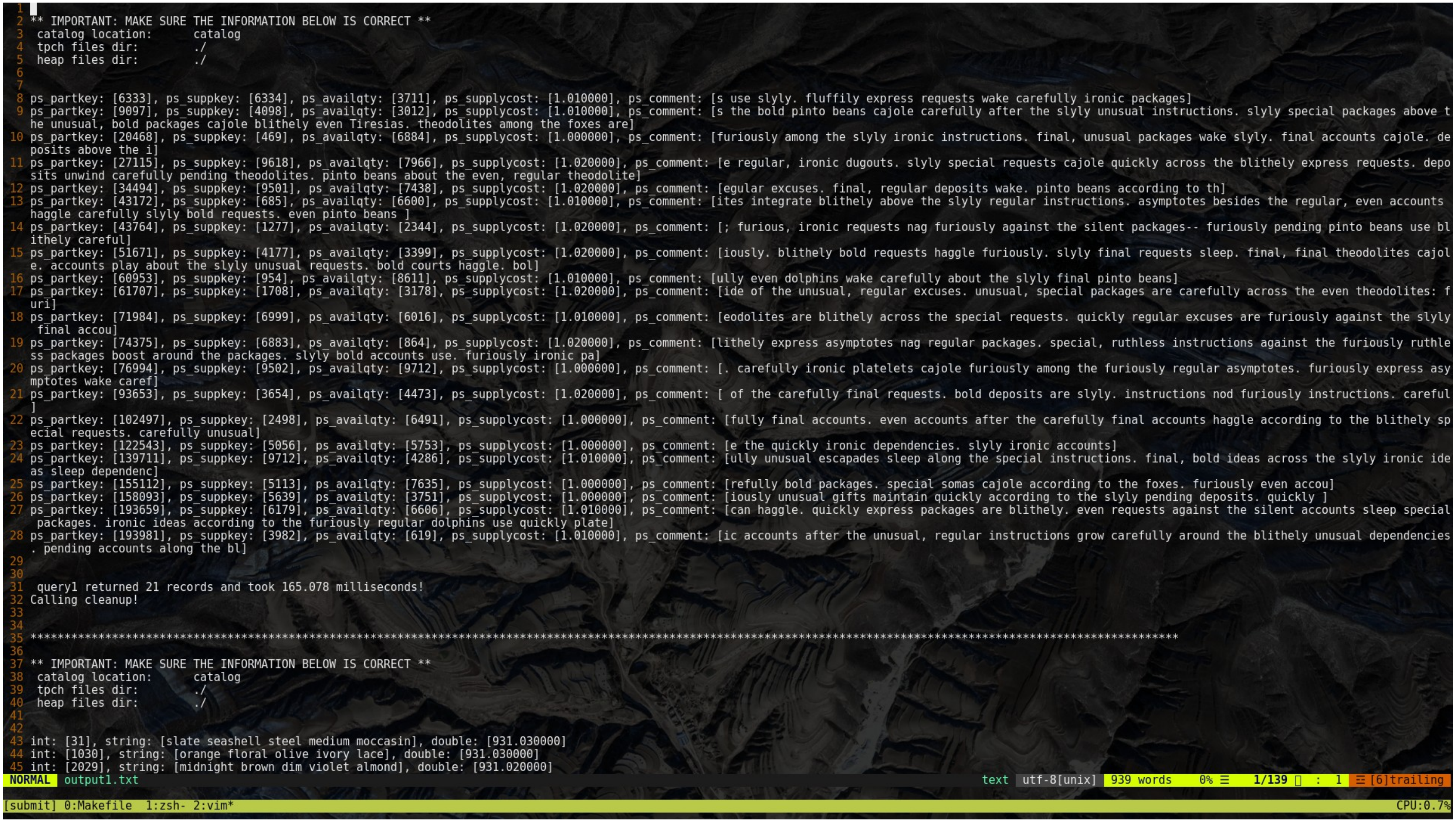
- void get_sum_rec(struct sum_args *arg, int *int_res, double *double_res): Function to sum the given records
- void *sum_thr(void *a): Function to create a thread for sum
- Sum :: Sum(): Constructor of the class to initialize initial argument
- Sum :: ~Sum(): Destructor of the class to destroy class variable
- void Sum :: Run(Pipe *in_pipe, Pipe *out_pipe, Function *func): Public function called from outside the class to start the Sum operation
- void Sum :: WaitUntilDone(): Function to pause future threads till current thread is completed
- void Sum :: Use_n_Pages(int n): Function to decide number of pages to be used

write_out.cc:

- void *write_thr(void *a): Function to create a thread for writing file
- WriteOut :: WriteOut(): Constructor of the class to initialize initial argument
- WriteOut :: ~WriteOut(): Destructor of the class to destroy class variable
- void WriteOut :: Run(Pipe *in_pipe, FILE *f, Schema *sch): Public function called from outside the class to start the writing file
- void WriteOut :: WaitUntilDone(): Function to pause future threads till current thread is completed
- void WriteOut :: Use_n_Pages(int n): Function to decide number of pages to be used

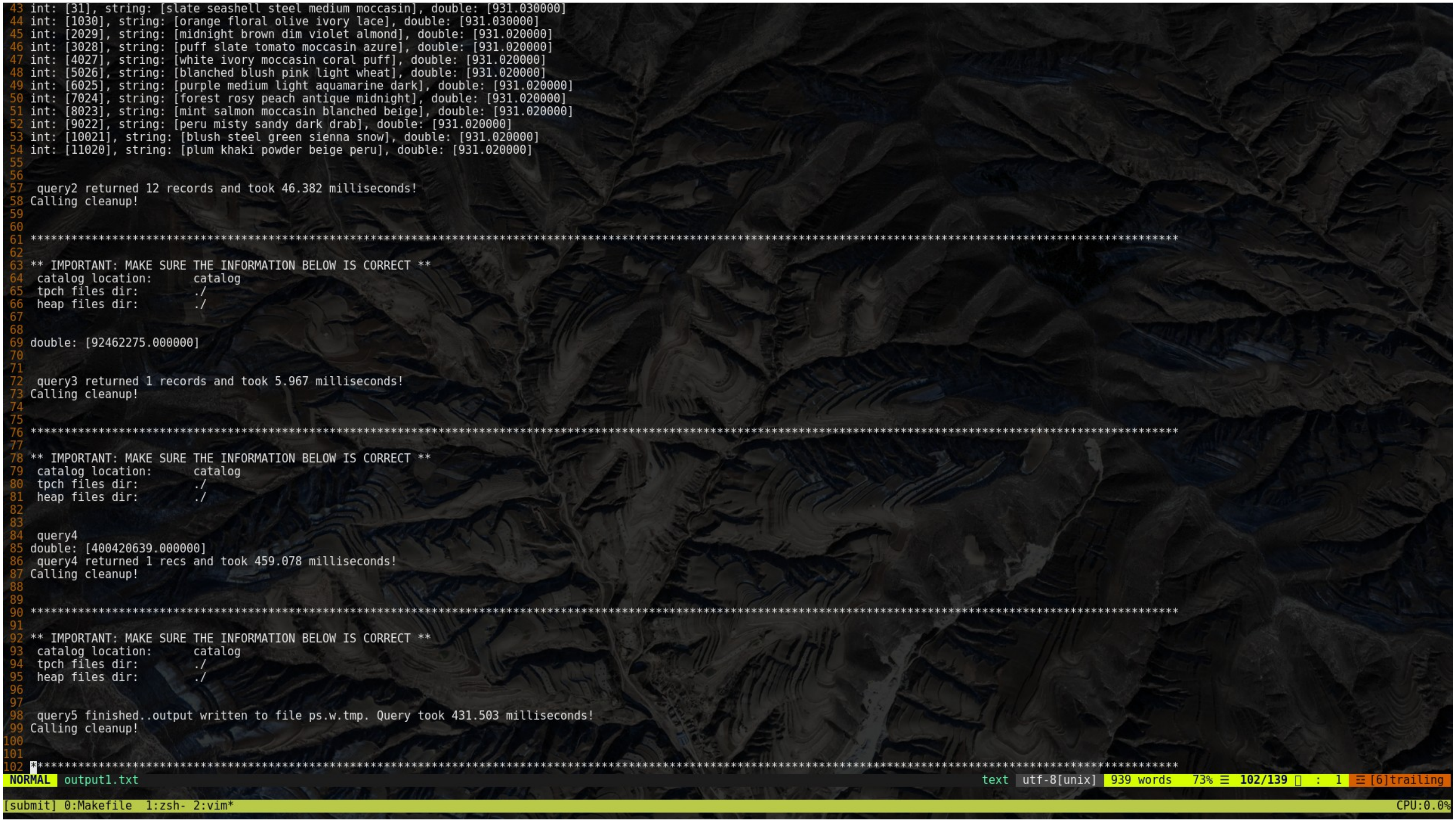
Outputs:

output1.txt
1 of 3



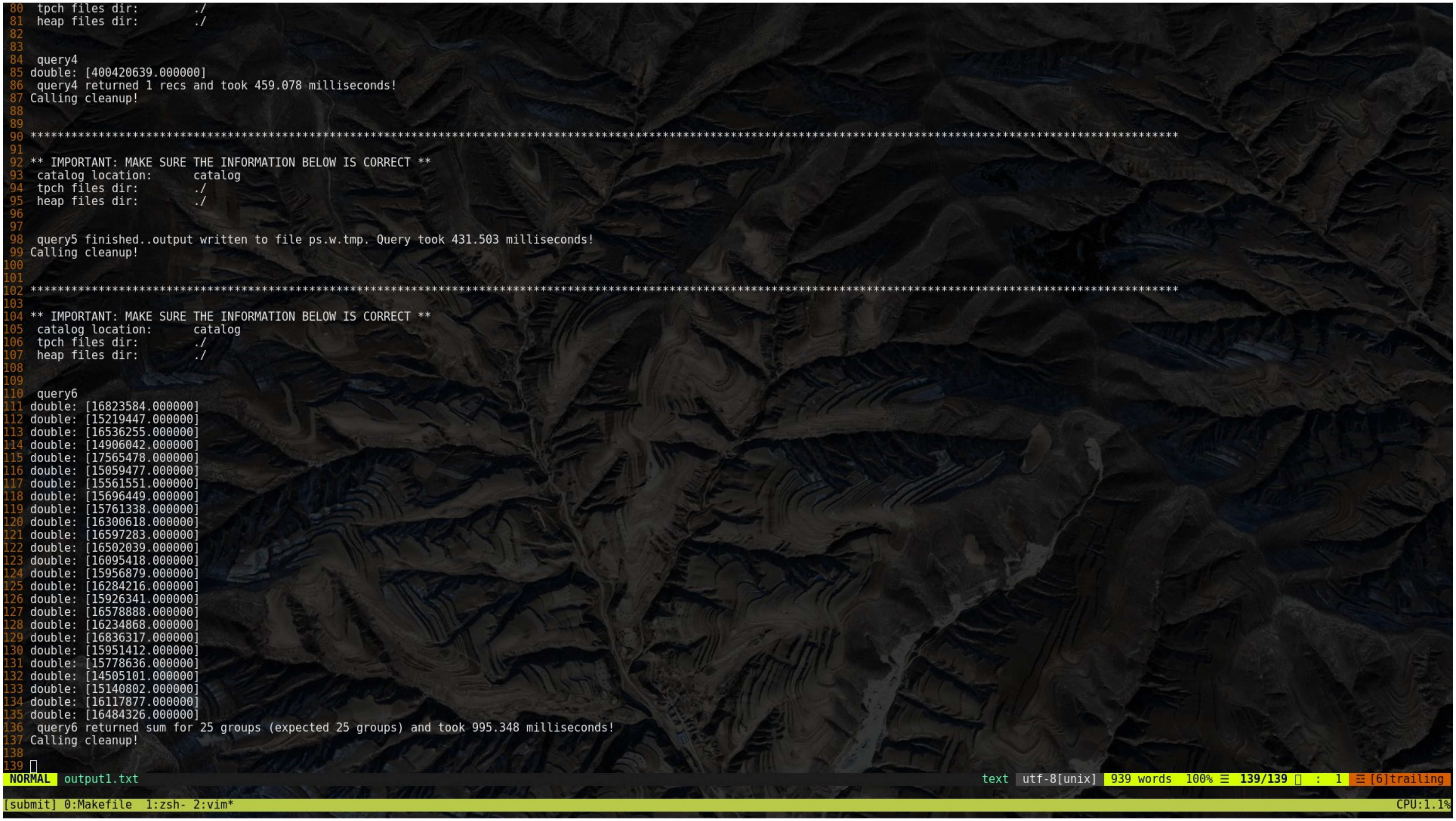
```

** IMPORTANT: MAKE SURE THE INFORMATION BELOW IS CORRECT **
 catalog location:      catalog
 tpch files dir:        ./
 heap files dir:        ./


ps_partkey: [6333], ps_suppkey: [6334], ps_availqty: [3711], ps_supplycost: [1.010000], ps_comment: [s use slyly. fluffily express requests wake carefully ironic packages]
ps_partkey: [9097], ps_suppkey: [4098], ps_availqty: [3012], ps_supplycost: [1.010000], ps_comment: [s the bold pinto beans cajole carefully after the slyly unusual instructions. slyly special packages above t
he unusual, bold packages cajole blithely even Tiresias. theodolites among the foxes are]
ps_partkey: [20468], ps_suppkey: [469], ps_availqty: [6884], ps_supplycost: [1.000000], ps_comment: [furiously among the slyly ironic instructions. final, unusual packages wake slyly. final accounts cajole. de
posits above the i]
ps_partkey: [27115], ps_suppkey: [9618], ps_availqty: [7966], ps_supplycost: [1.020000], ps_comment: [e regular, ironic dugouts. slyly special requests cajole quickly across the blithely express requests. depo
sits unwind carefully pending theodolites. pinto beans about the even, regular theodolite]
ps_partkey: [34494], ps_suppkey: [9501], ps_availqty: [7438], ps_supplycost: [1.020000], ps_comment: [egular excuses. final, regular deposits wake. pinto beans according to th]
ps_partkey: [43172], ps_suppkey: [685], ps_availqty: [6600], ps_supplycost: [1.010000], ps_comment: [ites integrate blithely above the slyly regular instructions. asymptotes besides the regular, even accounts
haggle carefully slyly bold requests. even pinto beans ]
ps_partkey: [43764], ps_suppkey: [1277], ps_availqty: [2344], ps_supplycost: [1.020000], ps_comment: [; furious, ironic requests nag furiously against the silent packages-- furiously pending pinto beans use bl
ithely careful]
ps_partkey: [51671], ps_suppkey: [4177], ps_availqty: [3399], ps_supplycost: [1.020000], ps_comment: [iously. blithely bold requests haggle furiously. slyly final requests sleep. final, final theodolites cajol
e. accounts play about the slyly unusual requests. bold courts haggle. bol]
ps_partkey: [60953], ps_suppkey: [954], ps_availqty: [8611], ps_supplycost: [1.010000], ps_comment: [ully even dolphins wake carefully about the slyly final pinto beans]
ps_partkey: [61707], ps_suppkey: [1708], ps_availqty: [3178], ps_supplycost: [1.020000], ps_comment: [ide of the unusual, regular excuses. unusual, special packages are carefully across the even theodolites: f
uri]
ps_partkey: [71984], ps_suppkey: [6999], ps_availqty: [6016], ps_supplycost: [1.010000], ps_comment: [eodolites are blithely across the special requests. quickly regular excuses are furiously against the slyly
 final accou]
ps_partkey: [74375], ps_suppkey: [6883], ps_availqty: [864], ps_supplycost: [1.020000], ps_comment: [lithely express asymptotes nag regular packages. special, ruthless instructions against the furiously ruthle
ss packages boost around the packages. slyly bold accounts use. furiously ironic pa]
ps_partkey: [76994], ps_suppkey: [9502], ps_availqty: [9712], ps_supplycost: [1.000000], ps_comment: [. carefully ironic platelets cajole furiously among the furiously regular asymptotes. furiously express asy
mptotes wake caref]
ps_partkey: [93653], ps_suppkey: [3654], ps_availqty: [4473], ps_supplycost: [1.020000], 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: [1.000000], ps_comment: [fully final accounts. even accounts after the carefully final accounts haggle according to the blithely sp
ecial requests. carefully unusual]
ps_partkey: [122543], ps_suppkey: [5056], ps_availqty: [5753], ps_supplycost: [1.000000], ps_comment: [e the quickly ironic dependencies. slyly ironic accounts]
ps_partkey: [139711], ps_suppkey: [9712], ps_availqty: [4286], ps_supplycost: [1.010000], ps_comment: [ully unusual escapades sleep along the special instructions. final, bold ideas across the slyly ironic ide
as sleep dependenc]
ps_partkey: [155112], ps_suppkey: [5113], ps_availqty: [7635], ps_supplycost: [1.000000], 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: [1.000000], 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: [1.010000], ps_comment: [can haggle. quickly express packages are blithely. even requests against the silent accounts sleep special
 packages. ironic ideas according to the furiously regular dolphins use quickly plate]
ps_partkey: [193981], ps_suppkey: [3982], ps_availqty: [619], ps_supplycost: [1.010000], ps_comment: [ic accounts after the unusual, regular instructions grow carefully around the blithely unusual dependencies
. pending accounts along the bl]


 query1 returned 21 records and took 165.078 milliseconds!
Calling cleanup!


********************************************************************************************************************************************************************************

** IMPORTANT: MAKE SURE THE INFORMATION BELOW IS CORRECT **
 catalog location:      catalog
 tpch files dir:        ./
 heap files dir:        ./


int: [31], string: [slate seashell steel medium moccasin], double: [931.030000]
int: [1030], string: [orange floral olive ivory lace], double: [931.030000]
int: [2029], string: [midnight brown dim violet almond], double: [931.020000]
 NORMAL  output1.txt                                                                                                          text  utf-8[unix]  939 words    0% ☰   1/139  :  1  ☰ [6]trailing

[submit] 0:Makefile  1:zsh- 2:vim*                                                                                                                                                       CPU:0.7%
```

output1.txt
2 of 3



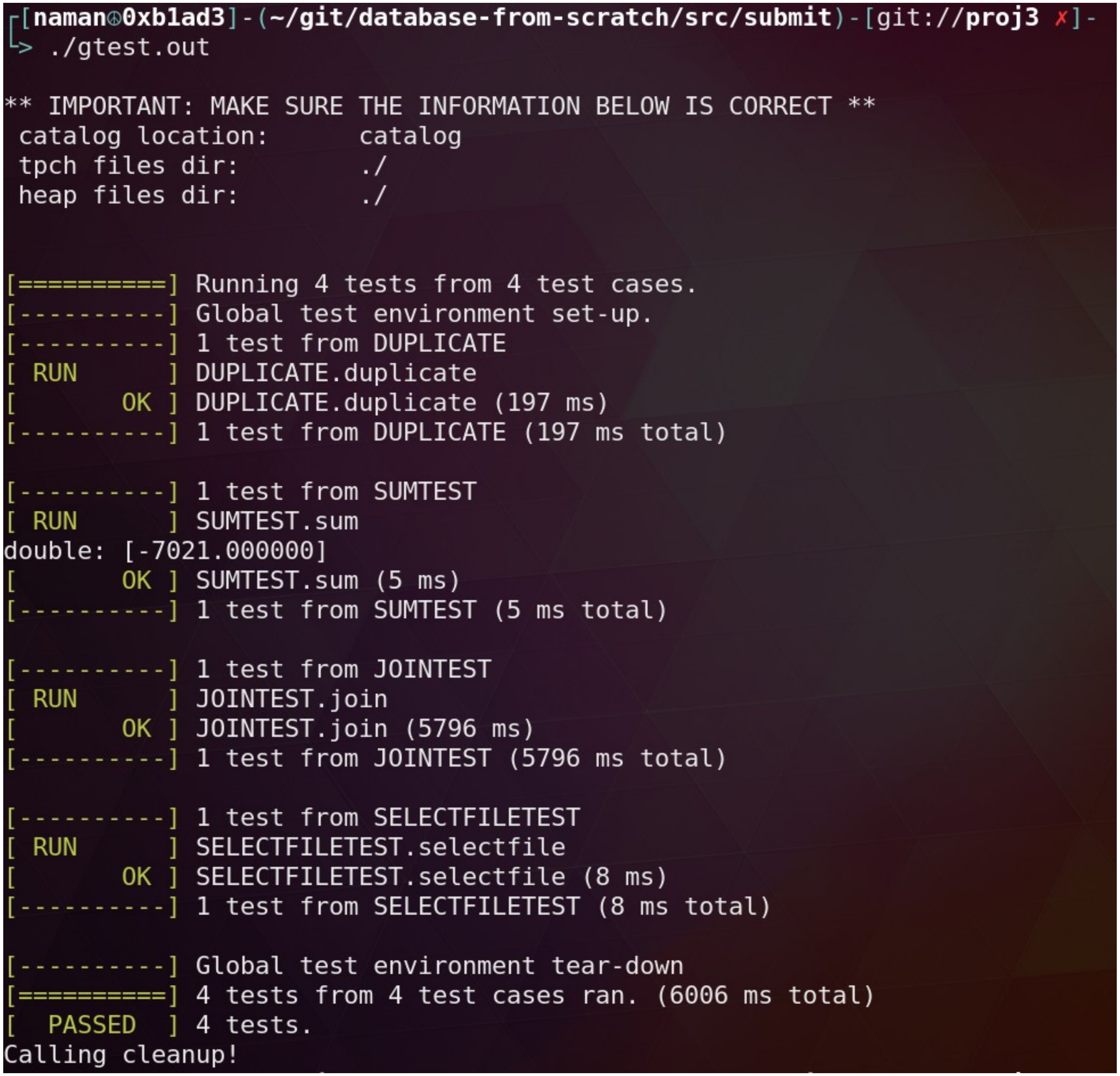
```
int: [31], string: [slate seashell steel medium moccasin], double: [931.030000]
int: [1030], string: [orange floral olive ivory lace], double: [931.030000]
int: [2029], string: [midnight brown dim violet almond], double: [931.020000]
int: [3028], string: [puff slate tomato moccasin azure], double: [931.020000]
int: [4027], string: [white ivory moccasin coral puff], double: [931.020000]
int: [5026], string: [blanched blush pink light wheat], double: [931.020000]
int: [6025], string: [purple medium light aquamarine dark], double: [931.020000]
int: [7024], string: [forest rosy peach antique midnight], double: [931.020000]
int: [8023], string: [mint salmon moccasin blanched beige], double: [931.020000]
int: [9022], string: [peru misty sandy dark drab], double: [931.020000]
int: [10021], string: [blush steel green sienna snow], double: [931.020000]
int: [11020], string: [plum khaki powder beige peru], double: [931.020000]


 query2 returned 12 records and took 46.382 milliseconds!
Calling cleanup!


******************************************************************************************************************************************************************************

** IMPORTANT: MAKE SURE THE INFORMATION BELOW IS CORRECT **
 catalog location:      catalog
 tpch files dir:        ./
 heap files dir:        ./


double: [92462275.000000]


 query3 returned 1 records and took 5.967 milliseconds!
Calling cleanup!


******************************************************************************************************************************************************************************

** IMPORTANT: MAKE SURE THE INFORMATION BELOW IS CORRECT **
 catalog location:      catalog
 tpch files dir:        ./
 heap files dir:        ./


 query4
double: [400420639.000000]
 query4 returned 1 recs and took 459.078 milliseconds!
Calling cleanup!


******************************************************************************************************************************************************************************

** IMPORTANT: MAKE SURE THE INFORMATION BELOW IS CORRECT **
 catalog location:      catalog
 tpch files dir:        ./
 heap files dir:        ./


 query5 finished..output written to file ps.w.tmp. Query took 431.503 milliseconds!
Calling cleanup!


******************************************************************************************************************************************************************************
 NORMAL  output1.txt                                                                                                                  text  utf-8[unix]  939 words   73% ☰  102/139  :  1  ☲ [6]trailing
[submit] 0:Makefile  1:zsh- 2:vim*                                                                                                                                                            CPU:0.0%
```

output1.txt
3 of 3



```
 tpch files dir:        ./
 heap files dir:        ./


 query4
double: [400420639.000000]
 query4 returned 1 recs and took 459.078 milliseconds!
Calling cleanup!


**************************************************************************************************************************************************************************

** IMPORTANT: MAKE SURE THE INFORMATION BELOW IS CORRECT **
 catalog location:      catalog
 tpch files dir:        ./
 heap files dir:        ./


 query5 finished..output written to file ps.w.tmp. Query took 431.503 milliseconds!
Calling cleanup!


**************************************************************************************************************************************************************************

** IMPORTANT: MAKE SURE THE INFORMATION BELOW IS CORRECT **
 catalog location:      catalog
 tpch files dir:        ./
 heap files dir:        ./


 query6
double: [16823584.000000]
double: [15219447.000000]
double: [16536255.000000]
double: [14906042.000000]
double: [17565478.000000]
double: [15059477.000000]
double: [15561551.000000]
double: [15696449.000000]
double: [15761338.000000]
double: [16300618.000000]
double: [16597283.000000]
double: [16502039.000000]
double: [16095418.000000]
double: [15956879.000000]
double: [16284216.000000]
double: [15926341.000000]
double: [16578888.000000]
double: [16234868.000000]
double: [16836317.000000]
double: [15951412.000000]
double: [15778636.000000]
double: [14505101.000000]
double: [15140802.000000]
double: [16117877.000000]
double: [16484326.000000]
 query6 returned sum for 25 groups (expected 25 groups) and took 995.348 milliseconds!
Calling cleanup!


 NORMAL  output1.txt                                                                                                                text  utf-8[unix]  939 words  100% ☰  139/139   :  1  ☲ [6]trailing
[submit] 0:Makefile  1:zsh- 2:vim*                                                                                                                                                      CPU:1.1%
```

GTest



```
┌[naman☮0xblad3]-(~/git/database-from-scratch/src/submit)-[git://proj3 ✗]-
└> ./gtest.out

** IMPORTANT: MAKE SURE THE INFORMATION BELOW IS CORRECT **
 catalog location:      catalog
 tpch files dir:        ./
 heap files dir:        ./


[==========] Running 4 tests from 4 test cases.
[----------] Global test environment set-up.
[----------] 1 test from DUPLICATE
[ RUN      ] DUPLICATE.duplicate
[       OK ] DUPLICATE.duplicate (197 ms)
[----------] 1 test from DUPLICATE (197 ms total)

[----------] 1 test from SUMTEST
[ RUN      ] SUMTEST.sum
double: [-7021.000000]
[       OK ] SUMTEST.sum (5 ms)
[----------] 1 test from SUMTEST (5 ms total)

[----------] 1 test from JOINTEST
[ RUN      ] JOINTEST.join
[       OK ] JOINTEST.join (5796 ms)
[----------] 1 test from JOINTEST (5796 ms total)

[----------] 1 test from SELECTFILETEST
[ RUN      ] SELECTFILETEST.selectfile
[       OK ] SELECTFILETEST.selectfile (8 ms)
[----------] 1 test from SELECTFILETEST (8 ms total)

[----------] Global test environment tear-down
[==========] 4 tests from 4 test cases ran. (6006 ms total)
[  PASSED  ] 4 tests.
Calling cleanup!
```