Database System Implementation
Project -1: Heap Data Base File Implementation

Project Members:
Srinadh Kakkera (UFID:0514-0863)
Spandana Edupuganti (UFID: 1733-6715)

Instructions to Run Code:
Change the path specified in the main.cc file and test.cc and Gtest.cc file to the current working directory.
$ make main
$ ./main

Instruction to run test Queries:
$make main
$make test.out
$./Test.out

Instructions to run GTests:
$ make main
$ make GTest.out
$./GTest.out

Method Description of HeapImplementation Class:

A DBFile instance is created in the DBFile.cc class and all of the following methods are implemented in the HeapImplementation Class.

void MoveFirst ()
This method checks whether the current page pointer points to the first record in the file. If it doesn't due to record retrievals,the function makes the pointer point to the first record of the file.

void Add (Record &rec)
This method takes the input record and puts it into the file.

int GetNext (Record &fetchMe)
The method gets the next record from the file which means the next of the record to which the pointer is currently pointing to and returns this record to the user and increments the pointer. This method returns an integer value which is zero in case when the function does not return any valid record which usually happens after the last record of the file has been returned.

int GetNext (Record &fetchMe, CNF &applyMe, Record &literal)
This method accepts CNF as a selection parameter and returns the next record which satisfies this parameter. A literal record is also created after parse tree for the CNF is processed and is used to check the selection parameter.

int Create (char *f_path, f_Type my_Type, void *startup)
This method creates a file with the name which is passed as first parameter in the DB and of the type passed as the second parameter(type may be heap, sorted or a tree). It returns 1 if success and 0 on failure.

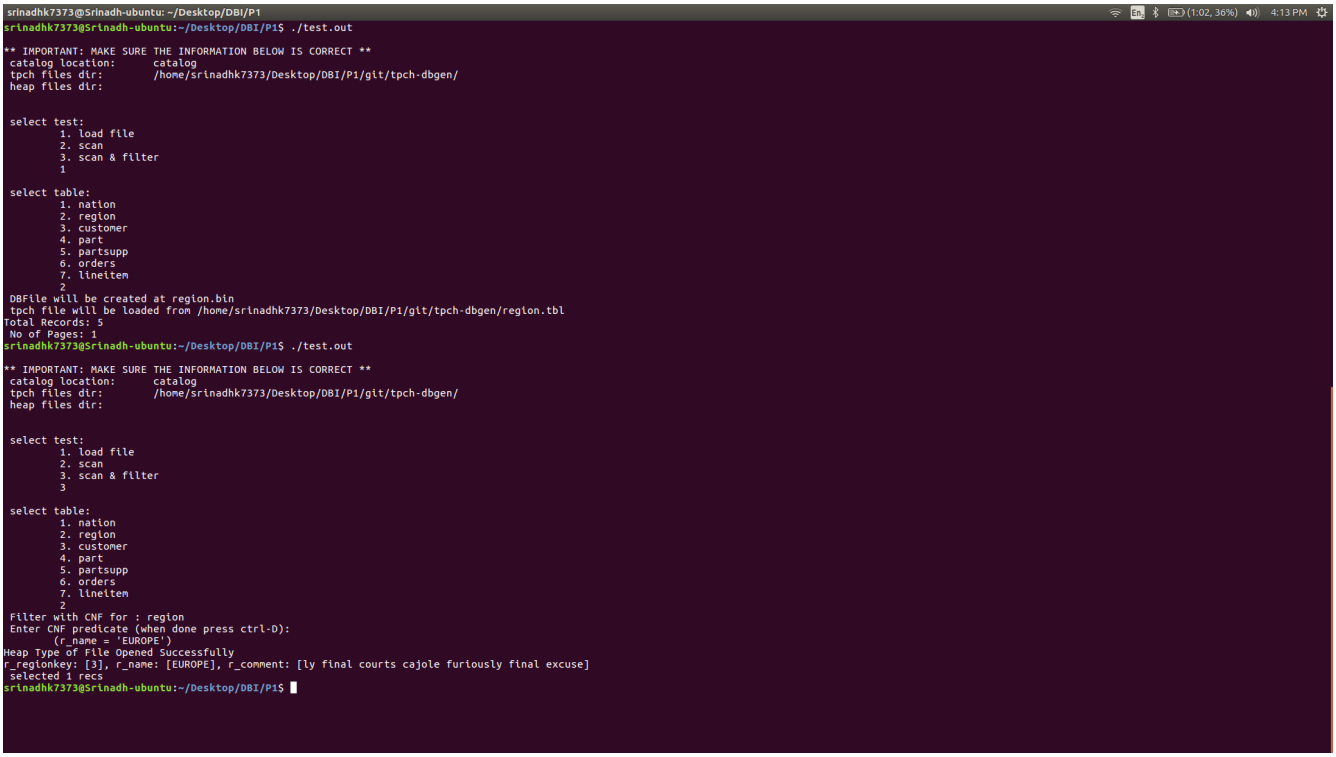
int Open (char *f_path)
This method looks for a file with the name passed as a parameter in the Dbfile and opens this file. It returns value 1 on success and zero on failure.

int Close ()
This method just closes the file and returns 1 on success and 0 on failure.

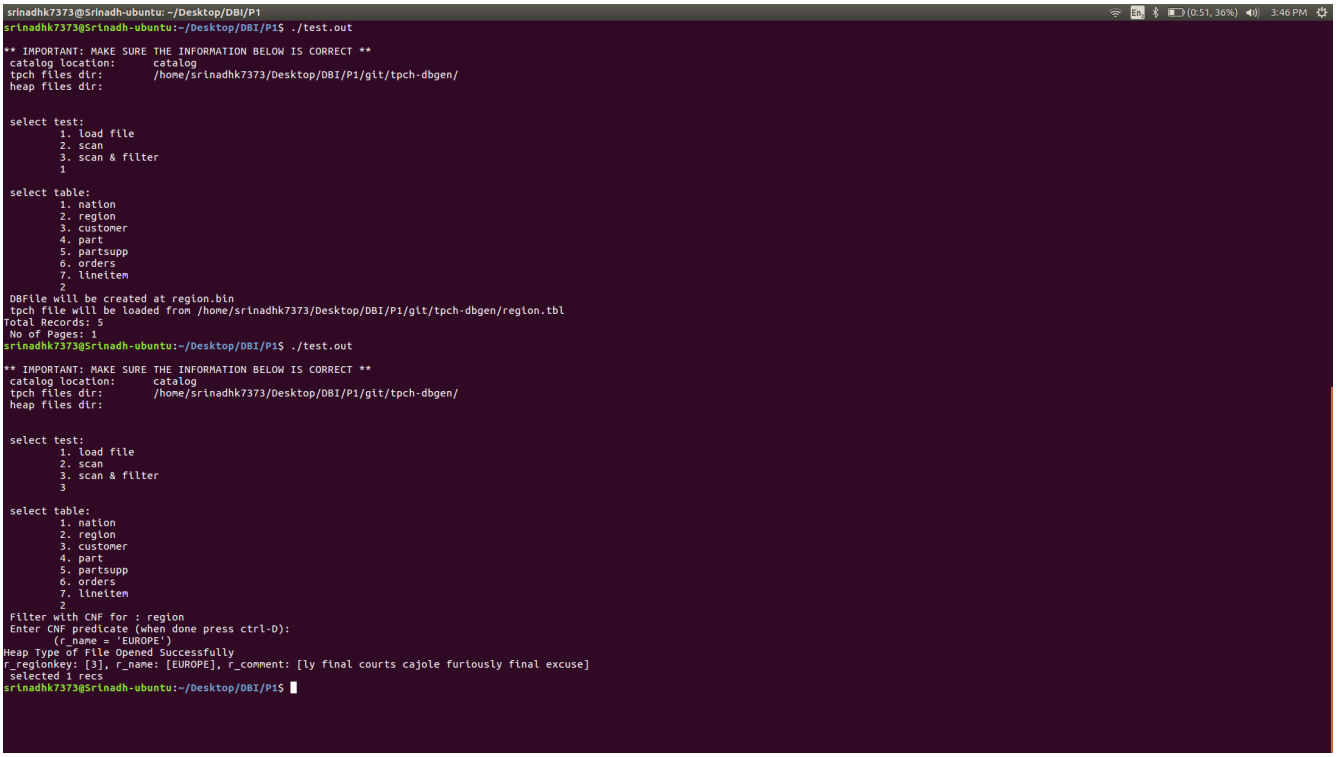
void Load (Schema &f_schema, const char *loadpath)
This method loads the bulk amount of data from the data file whose name is the paramter passed to the method and it appends the additional data to this by fetching from the Record.h using the method SuckNextRecord().

Screenshots of the output:

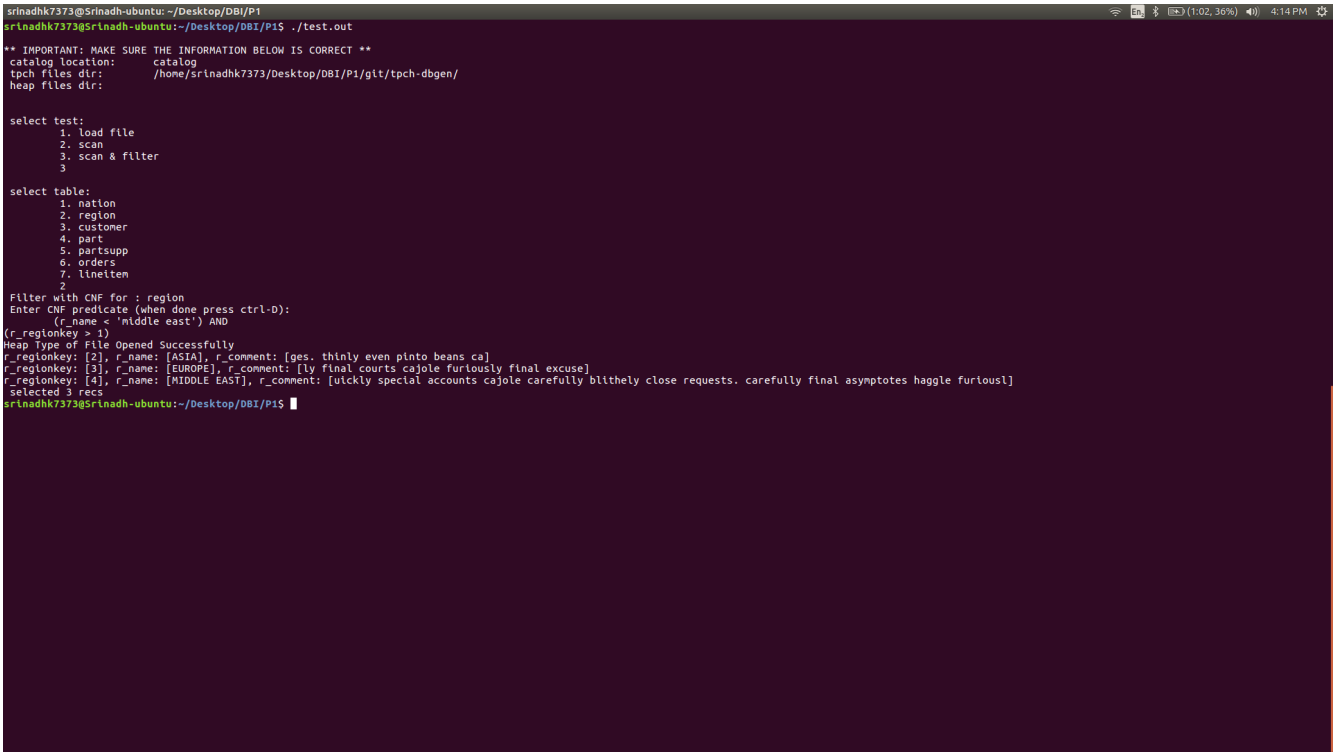
Query1: for 10MB data
(r_name = 'EUROPE')



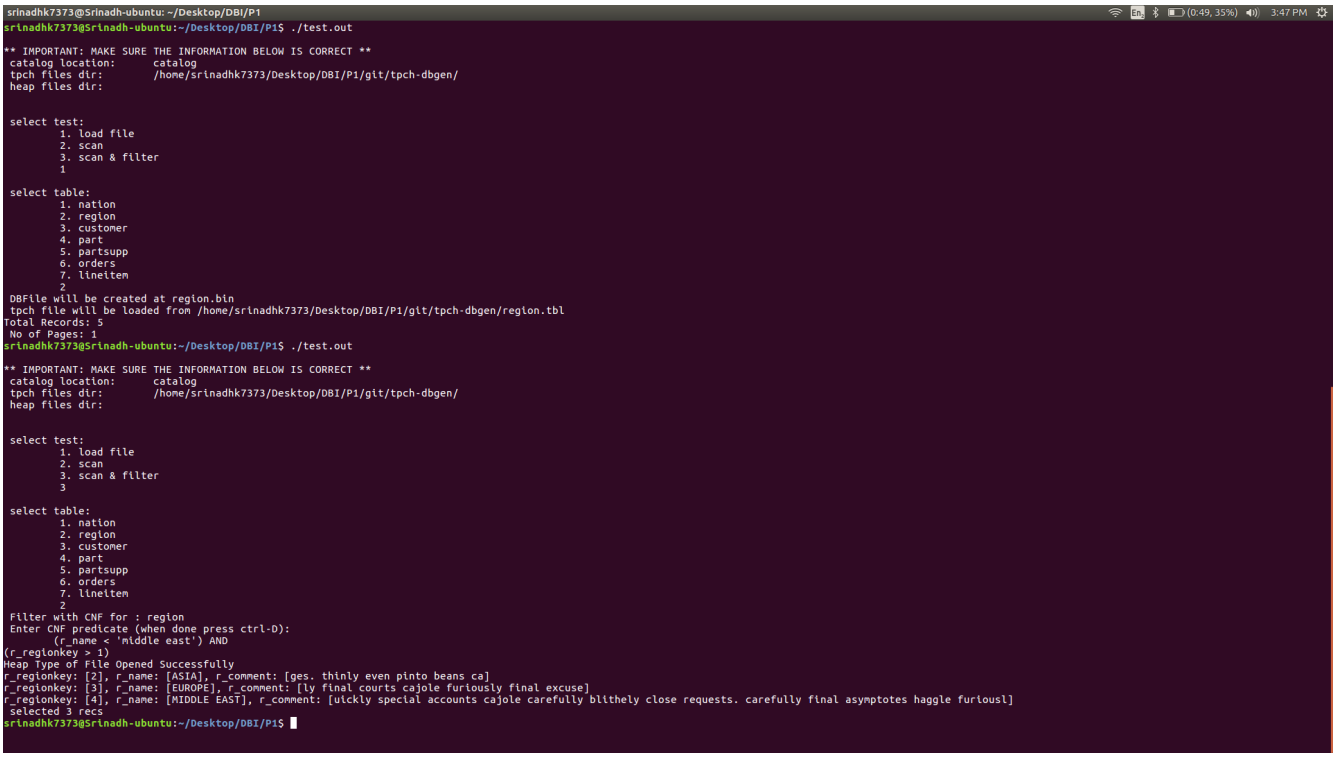
Query 1: for 1 GB data
(r_name = 'EUROPE')



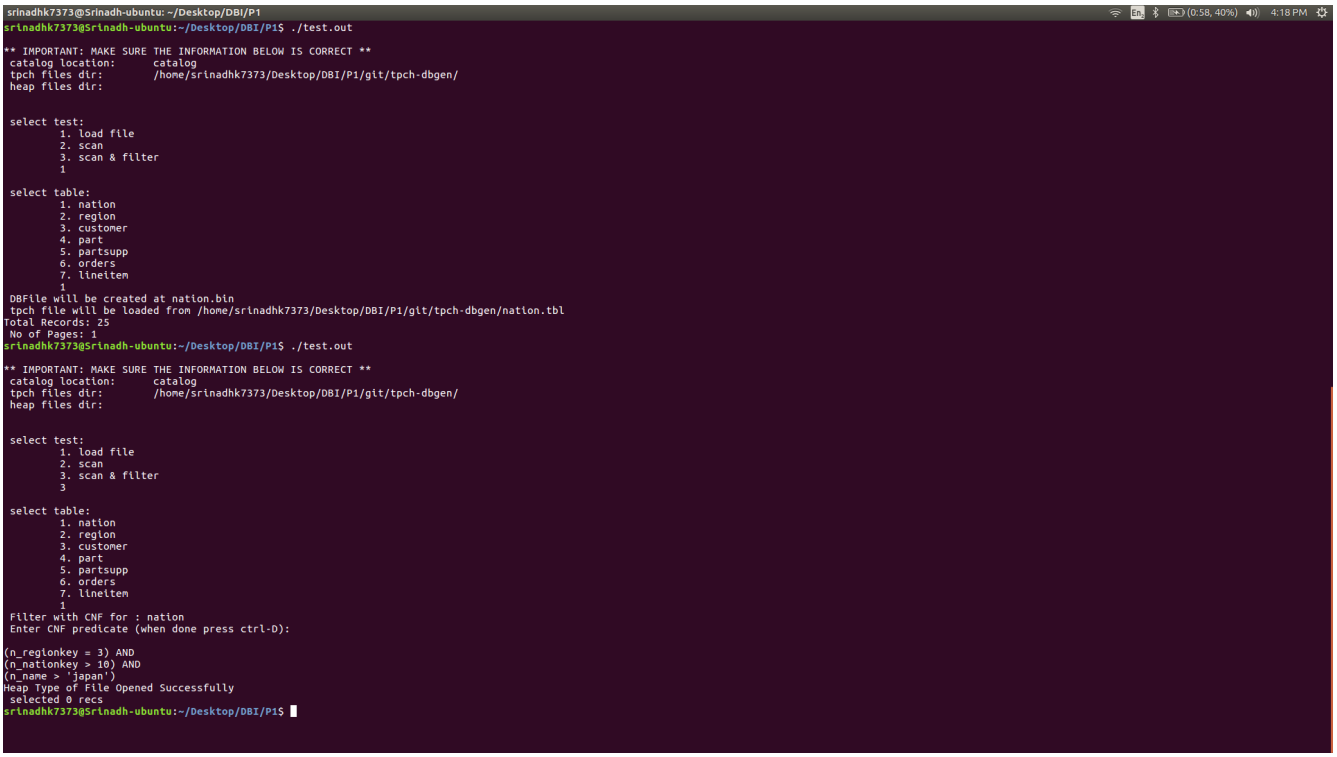
Query 2 : for 10 MB data
(r_name < 'middle east') AND (r_regionkey > 1)



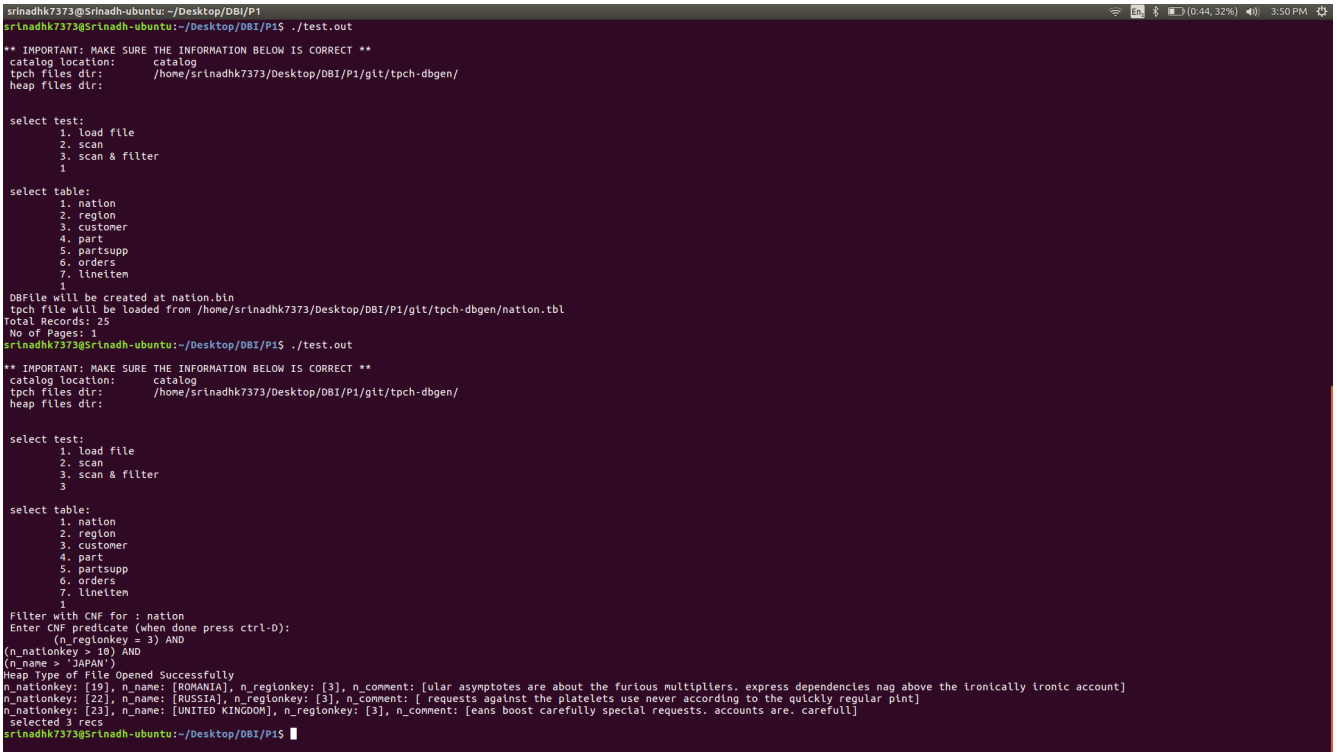
Query 2: for 1GB data
(r_name < 'middle east') AND (r_regionkey > 1)



Query 3: for 10 MB data
(n_regionkey = 3) AND (n_nationkey > 10) AND (n_name > 'japan')



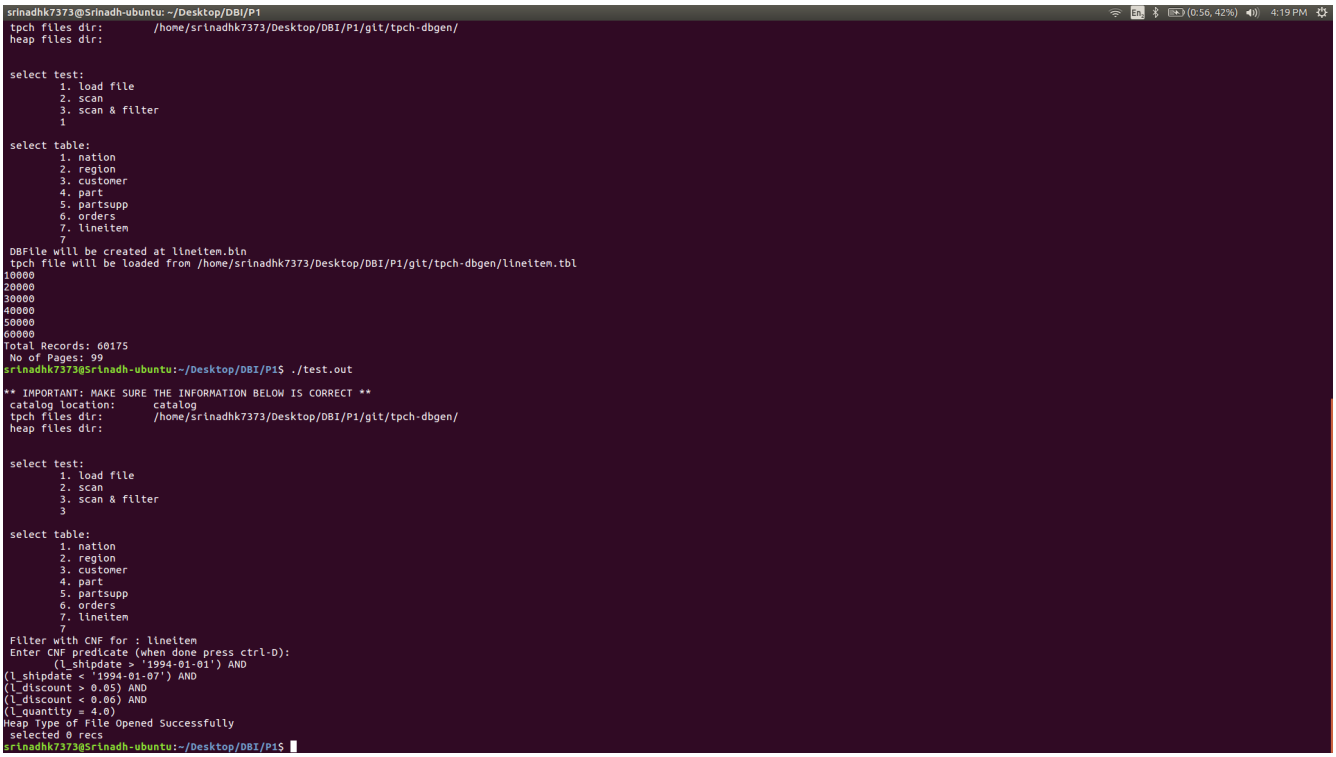
Query 3: for 1 GB data
(n_regionkey = 3) AND (n_nationkey > 10) AND (n_name > 'japan')



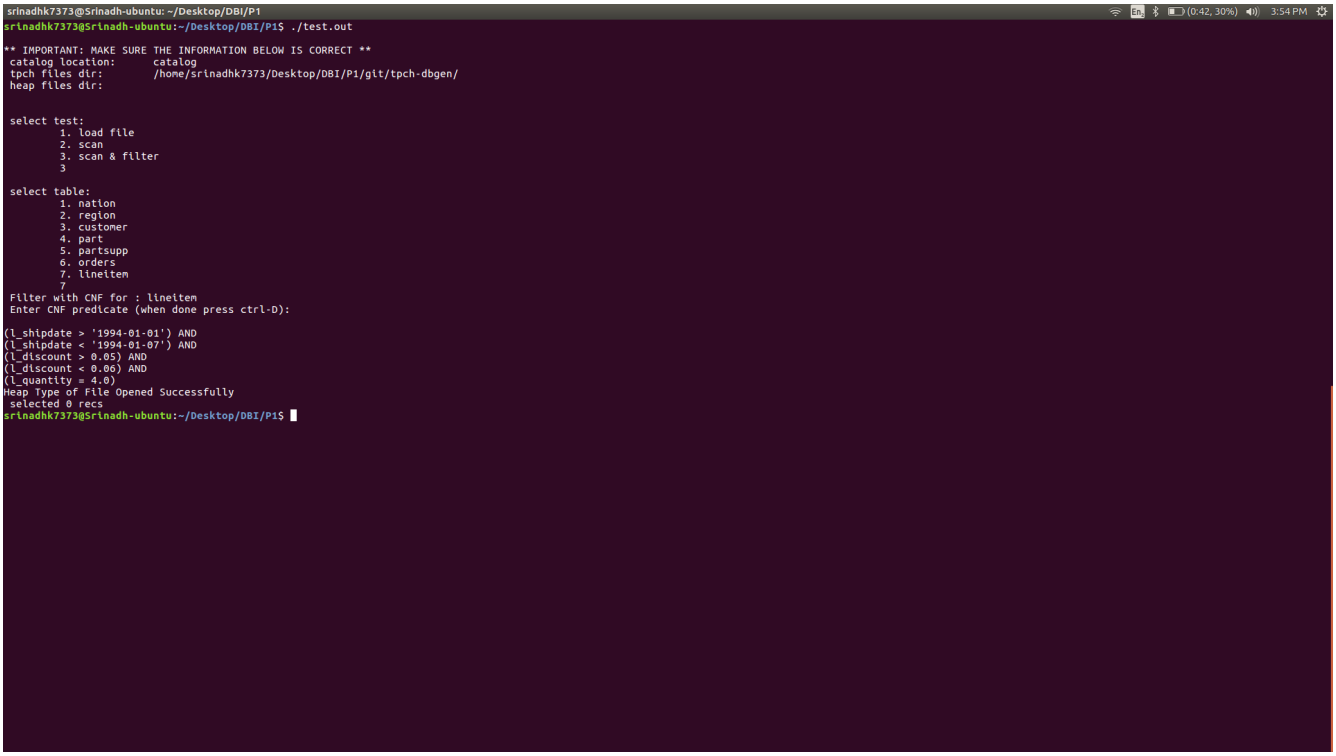
Query 11: for 10MB data
(l_shipdate > '1994-01-01') AND

(l_shipdate < '1994-01-07') AND
(l_discount > 0.05) AND
(l_discount < 0.06) AND

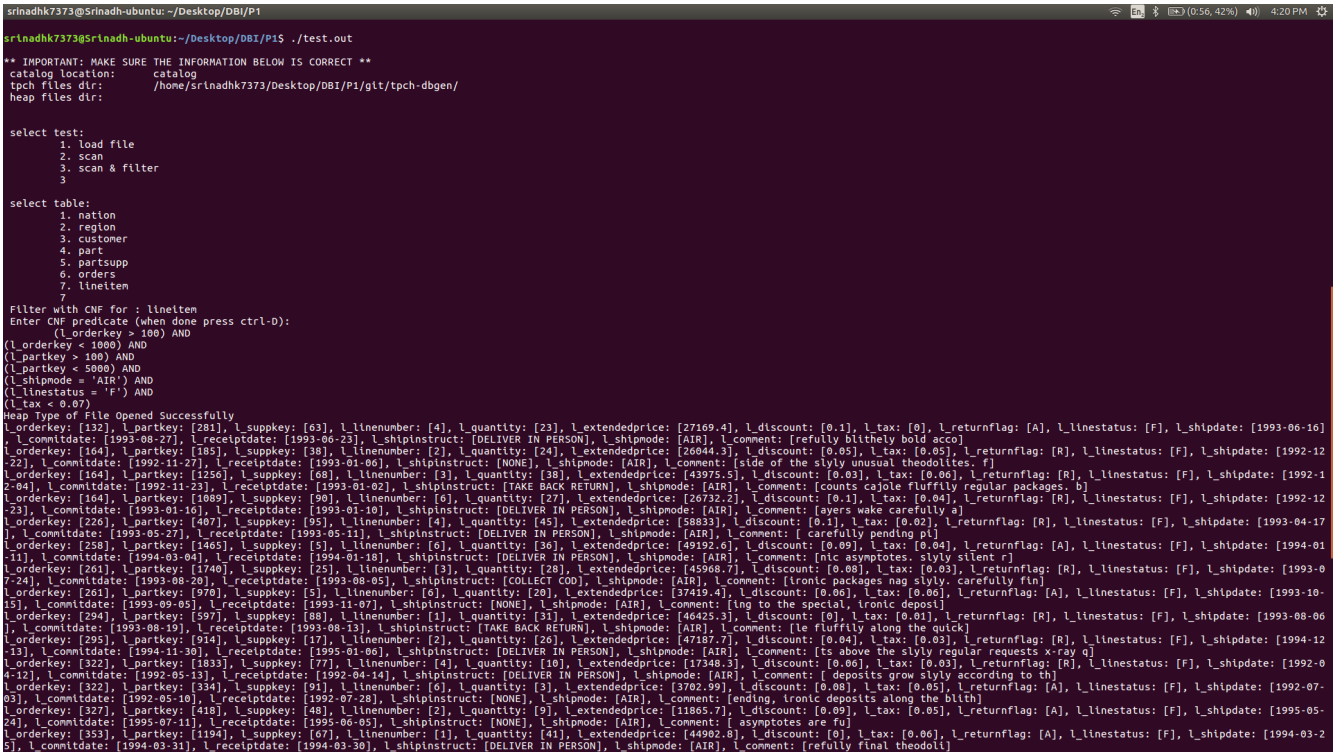
(l_quantity = 4)

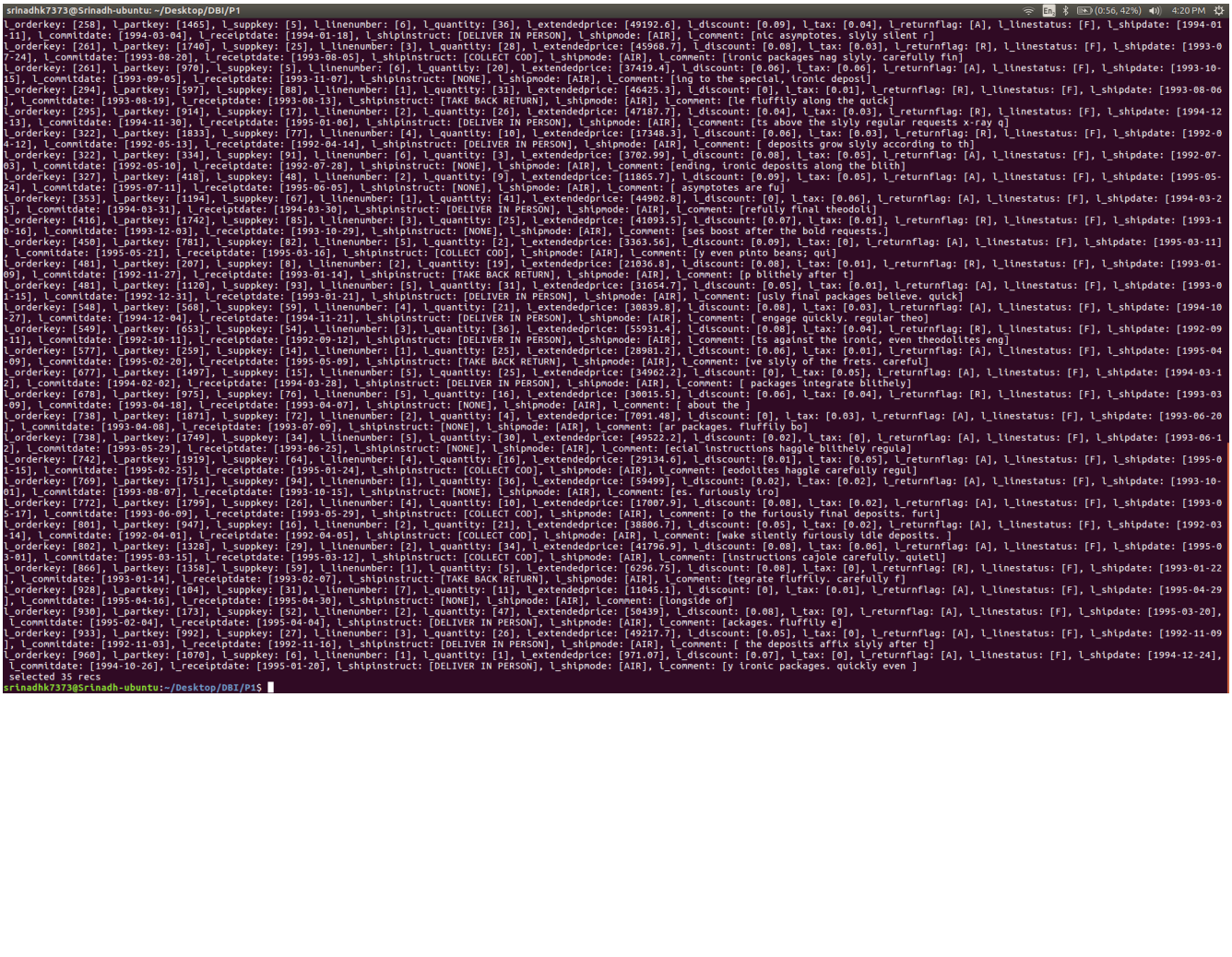


Query 11: for 1 GB data
(l_shipdate > '1994-01-01') AND
(l_shipdate < '1994-01-07') AND
(l_discount > 0.05) AND
(l_discount < 0.06) AND
(l_quantity = 4)

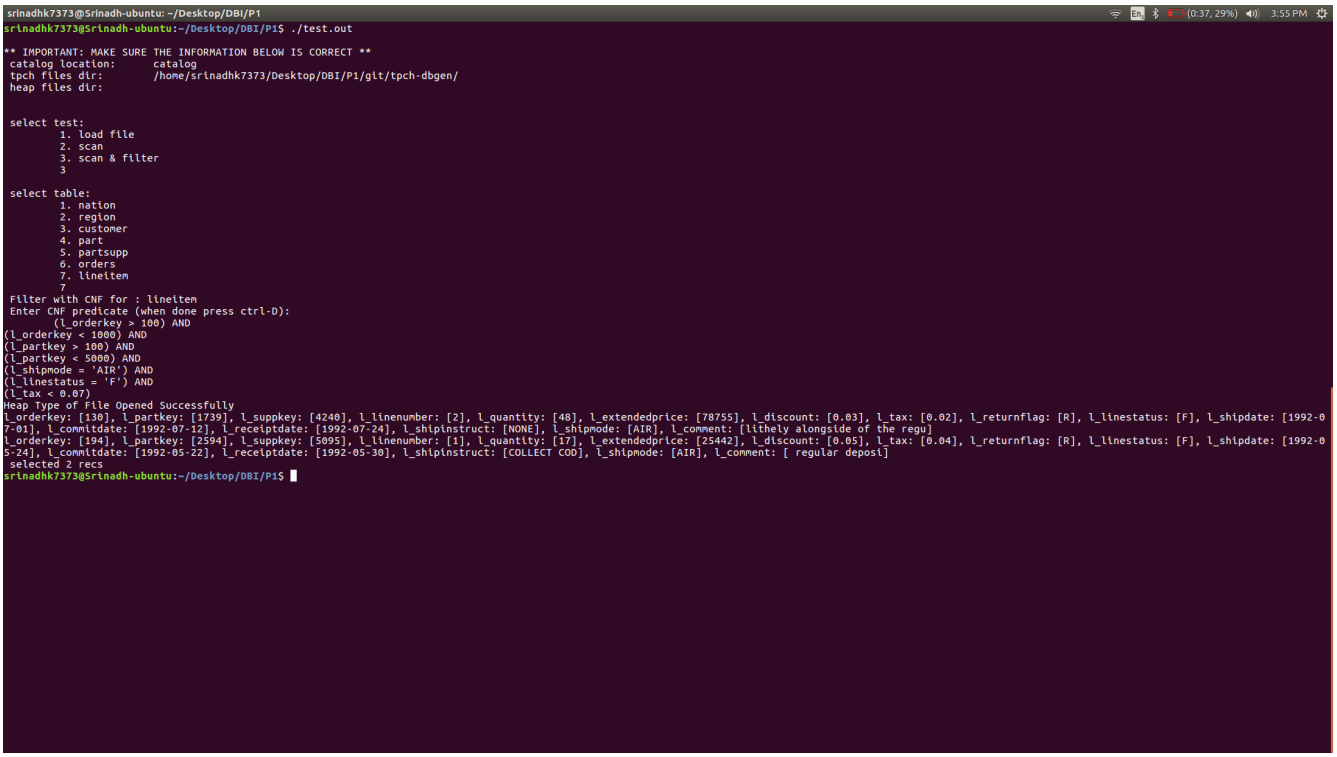


Query 12: for 10MB data
(l_orderkey > 100) AND
(l_orderkey < 1000) AND
(l_partkey > 100) AND
(l_partkey < 5000) AND
(l_shipmode = 'AIR') AND
(l_linestatus = 'F') AND
(l_tax < 0.07)


srinadhk7373@Srinadh-ubuntu: ~/Desktop/DBI/P1
srinadhk7373@Srinadh-ubuntu:~/Desktop/DBI/P1$ ./test.out
** IMPORTANT: MAKE SURE THE INFORMATION BELOW IS CORRECT **
catalog location: catalog
tpch files dir: /home/srinadhk7373/Desktop/DBI/P1/git/tpch-dbgen/
heap files dir:
select test:
1. load file
2. scan
3. scan & filter
3
select table:
1. nation
2. region
3. customer
4. part
5. partsupp
6. orders
7. lineitem
7
Filter with CNF for : lineitem
Enter CNF predicate (when done press ctrl-D):
(l_orderkey > 100) AND
(l_orderkey < 1000) AND
(l_partkey > 100) AND
(l_partkey < 5000) AND
(l_shipmode = 'AIR') AND
(l_linestatus = 'F') AND
(l_tax < 0.07)
Heap Type of File Opened Successfully


srinadhk7373@Srinadh-ubuntu: ~/Desktop/DBI/P1
selected 35 recs
srinadhk7373@Srinadh-ubuntu:~/Desktop/DBI/P1$

Query 12 : for 1 GB data
(l_orderkey > 100) AND
(l_orderkey < 1000) AND
(l_partkey > 100) AND
(l_partkey < 5000) AND
(l_shipmode = 'AIR') AND
(l_linestatus = 'F') AND
(l_tax < 0.07)



Method Description of the GTest.cc class:

TEST(DBFileTest, Create_Equal)
This method tests whether the file is created.

TEST(DBFileTest, Create_NotEqual)
This method tests whether program is exiting when an invalid file is asked to create.

TEST(DBFileTest, Open_Equal)
This method tests whether file is opened successfully.

TEST(DBFileTest, Open_NotEqual)
This method tests whether all invalid file open scenarios are handled.

TEST(DBFileTest, Close_nation_DBFile)
This method tests whether the given file is closed successfully.

TEST(DBFileTest, Close_Empty_File)
This method tests whether program handles the case where an empty file is asked to close.

Screenshots of GTest results: