

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 2 milestone 2. The heap and sorted file variation to DB file implementation has 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 the exception of exactly three, viz. bin/, tpch-dbgen/ and tmp/. The bin/ directory is the output destination for the dbfile, i.e. dbfile directory. The tpch-dbgen is the git repository for the TPCB sub program. tmp/ houses all the test case files, test{1,2,3}.txt. Please make sure the tpch-dbgen/*.tbl files are present BEFORE test is run.

Note that add_data and test1 functions have been modified in the test.cc source file. The former modification helps in not wasting one record from not being processed after end of each loop and returns the real value of "proc" variable. The latter modification helps in overcoming a bug related to input of query statement which led to infinite loop for run length specification. Thus, the sequence of input in test cases has also been modified accordingly. Run length is now provided AFTER the query.

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_p2-2.zip has one sub-directory, src/.

```
$> unzip NamanArora_NikunjSarda_p2-2.zip
```

```
$> cd proj2_2
```

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>
```

```
$> sync
```

All the available relation binary files are available in bin/

To clean the repository, not including files in bin/ and *.tbl files,

```
$> make clean
```

To clean the repository, including files in bin/ and tpch-dbgen/*.tbl:

```
$> make distclean
```

Change log:

- ➔ Updated dbfile.cc and dbfile.h to add an interface class that internally calls heap and sorted files.
- ➔ Updated file.cc and file.h to append the SortInfo, type of file and curLength of pages in the beginning page of the database file as metadata.
- ➔ Added heap.cc and heap.h as the implementation of Heap file completed in prior submissions.
- ➔ Updated OrderMaker class to include an overloaded assignment operator and a function to generate a query OrderMaker instance given another OrderMaker and a CNF.
- ➔ Updated test.cc, specifically add_data function so that the internal loop does not waste a sucked record when the proc flag has reached its limit. Also add_data all returns proc-1 since there is always extra record counted at the end of loop.
- ➔ Added sort_helper.cc, sorted.cc and sorted.h as implementation of Sorted File.
- ➔ enum fType, struct SortInfo and struct metadata are defined in defs.h

dbfile.cc:

- ➔ struct SortInfo &SortInfo :: operator= (const struct SortInfo &) :
SortInfo operator assignment operator overload.
- ➔ int DBFile :: get_type(const char *):
This opens up a dummy File class instance to read the metadata of file requested for opening.
Inputs the file name and outputs the type of type fType defined at defs.h
- ➔ int DBFile :: Create(const char *, ftype , SortInfo *):
Returns exit status, 0 if creation fails and 1 if it succeeds.
- ➔ int DBFile :: Open(const char *):
Returns 0 on failure and 1 on success.
- ➔ void DBFile :: Add(Record *):
Takes in a record in form Record * and adds it to currently opened file.
- ➔ void DBFile :: Load(Schema *, const char *):
Inputs a schema and file name of the table to load the records. Calls sorted or heap file functions accordingly.
- ➔ void DBFile :: MoveFirst():
Move to the very first record.
- ➔ int DBFile :: GetNext(Record *):
Gets the next record in sequential manner. Returns 0 on error and 1 on success.
- ➔ int DBFile :: GetNext(Record *, CNF *, Record *):
Gets the next record based on given CNF and literal (in case of sorted file).
- ➔ int DBFile :: Close():

Closes the file and returns the status flag, 0 if error 1 on success.

sorted.cc:

- ➔ `file_info :: file_info(struct SortInfo *, const char *, File *, Page *):`
Constructor of struct `file_info` used to pass file information between `SortedFile` and `SortedHelper` classes.
- ➔ `SortedFile :: SortedFile(struct SortInfo *, const char *, int):`
Sorted File constructor. Inputs the struct `SortInfo`, file name and a flag, if this instance is supposed to be pseudo sorted file. Used internally.
- ➔ `int SortedFile :: get_next_query(Record *fetchme, CNF *cnf, Record *literal):`
This is used when a query `OrderMaker` is either present already or has been successfully created. Returns 0 on failure and 1 on success.
- ➔ `int SortedFile :: get_next_no_query(Record *fetchme, CNF *cnf, Record *literal):`
This is used when `OrderMaker` query is not available and returns 1 on success and 0 on failure.
- ➔ `int SortedFile :: binary_search(Record *fetchme, CNF *cnf, Record *literal):`
This is implementation of binary search algorithm to search for records within the pages of files. Returns 0 on failure and 1 on success.

sort_helper.cc:

- ➔ `SortedHelper :: SortedHelper(struct file_info *f_info):`
This is constructor of `SortHelper` class and inputs the file information `FileInfo` structure.
- ➔ `int SortedHelper :: fetch(int pg_num, int pseudo):`
This is used to fetch a page defined by a number that is inputted. It takes pseudo flag input too so that it can pass on along to `unset_dirty` (See below).
- ➔ `void SortedHelper :: set_create(int val):`
This sets up the create flag for file to a desired value. 0 for create flag means the file is just created or was closed empty and 1 otherwise.
- ➔ `int SortedHelper :: get_create():`
Returns the value of create flag.
- ➔ `void SortedHelper :: set_dirty(int pseudo):`
This sets up the `BigQ` and `Pipe` classes and flips the dirty flag to 1.
- ➔ `int SortedHelper :: feed():`
This is used to feed the remaining records from file which were already written to it after `unset_dirty` is called. If none such records are present, it just closes the `in_pipe`.
- ➔ `int SortedHelper :: writeback(int flag):`
This has two modes for operation, one is writing to a temporary file and the other is leveraging underlying `File` class to actually write pages.
- ➔ `int SortedHelper :: unset_dirty(int pseudo):`
This unsets the dirty flag for the file after each write operation.
- ➔ `void SortedHelper :: Add(Record *placeholder, int internal):`
This adds a record to the file, depending on the flag, either to the `in_pipe` or actually to the `File`.

- ➔ `int SortedHelper :: fetch_curr_pg():`
Returns the current page number of file.
- ➔ `void SortedHelper :: set_curr_pg(int pg_num):`
Sets the current page number of the file to specified value.
- ➔ `int SortedHelper :: chk_dirty():`
Returns the current value of dirty.

heap.cc:

Heap file implementation, previously DBFile present in `dbfile.{cc,h}`, now shifted to `heap.{cc,h}`

comparison.cc:

- ➔ `OrderMaker &OrderMaker :: operator= (const OrderMaker &in):`
Assignment operator overload for the OrderMaker class. Useful in serializing it to File class.

file.cc:

- ➔ `struct metadata &operator= (struct metadata &)`
Assignment operator overload for metadata class, useful in serialization of metadata to File class.

Outputs:

Test case 1

```
End of records!Records: 146340
    added 390 recs..so far 146340
End of records!Records: 147045
    added 705 recs..so far 147045
End of records!Records: 147430
    added 385 recs..so far 147430
End of records!Records: 148053
    added 623 recs..so far 148053
End of records!Records: 148115
    added 62 recs..so far 148115
End of records!Records: 148298
    added 183 recs..so far 148298
End of records!Records: 148654
    added 356 recs..so far 148654
End of records!Records: 149431
    added 777 recs..so far 149431
End of records!Records: 149699
    added 268 recs..so far 149699
End of records!Records: 149965
    added 266 recs..so far 149965
End of records!Records: 150000
    added 35 recs..so far 150000

create finished.. 150000 recs inserted
Calling cleanup!

[naman@0xb1ad3] - (~/git/database-from-scratch/src/submit) - [git://mile2 x]-
[0] 0:vim- 1:zsh*
```


Test case 2

```
acctbal: [4923.4], c_mktsegment: [MACHINERY], c_comment: [ silent requests haggle carefully regular, regular packages. enticing, even deposits wake  
slyly among the]  
.c_custkey: [12022], c_name: [Customer#000012022], c_address: [PwVDMfqZtKW0o], c_nationkey: [24], c_phone: [34-998-408-4382], c_acctbal: [2877.36],  
c_mktsegment: [HOUSEHOLD], c_comment: [layers. regular pinto beans about the quickly regular ideas cajole blithely]  
.c_custkey: [125210], c_name: [Customer#000125210], c_address: [BJiJAg79TvSTdxgPeb63sUqKa0Vv], c_nationkey: [24], c_phone: [34-998-711-2597], c_acctbal: [6678.32], c_mktsegment: [FURNITURE], c_comment: [ly ironic, ironic accounts. theodolites among the slyly even patterns nag slyly fluffi]  
.c_custkey: [50667], c_name: [Customer#000050667], c_address: [9aGxwCYLcG6y2eA], c_nationkey: [24], c_phone: [34-999-115-2195], c_acctbal: [8764.52]  
], c_mktsegment: [FURNITURE], c_comment: [mptotes around the slyly bold instructions boost furi]  
.c_custkey: [7288], c_name: [Customer#000007288], c_address: [,dZCqJe94SqXQjxAd7WxRMYyFVKGD4pFIRAshk], c_nationkey: [24], c_phone: [34-999-121-7718]  
], c_acctbal: [82.07], c_mktsegment: [HOUSEHOLD], c_comment: [ns sublate furiously. carefully ironic platelets grow furiously. ironic, bold theodol  
ites hang above th]  
.c_custkey: [84642], c_name: [Customer#000084642], c_address: [MJerprOnUYRK], c_nationkey: [24], c_phone: [34-999-195-7029], c_acctbal: [6912.63],  
c_mktsegment: [BUILDING], c_comment: [totes above the pending, regular hockey players sleep blithely after the blithely even deposits. carefully re  
gula]  
.c_custkey: [148611], c_name: [Customer#000148611], c_address: [HoNNKAELd2vMNYGwpw3ZZgSNAdS7f0dNIH64], c_nationkey: [24], c_phone: [34-999-283-6448]  
], c_acctbal: [24.12], c_mktsegment: [HOUSEHOLD], c_comment: [requests above the blithely pending asymptotes sleep carefully about the slyly even e  
xcu]  
.c_custkey: [102689], c_name: [Customer#000102689], c_address: [uPFxngvS0aDQIL], c_nationkey: [24], c_phone: [34-999-363-7145], c_acctbal: [6256.97]  
], c_mktsegment: [AUTOMOBILE], c_comment: [ly regular asymptotes haggle fluffily according to the furiously bold courts. daring ]  
.c_custkey: [35478], c_name: [Customer#000035478], c_address: [0t5zcyiiLhzXXivX4Y2FeToQIh,i0vs6zyA], c_nationkey: [24], c_phone: [34-999-379-6016],  
c_acctbal: [6097.18], c_mktsegment: [MACHINERY], c_comment: [ithout the fluffily ironic packages. daring, even de]  
.c_custkey: [103505], c_name: [Customer#000103505], c_address: [XPNXpdCD7HRqcV8ABpjF3ixDc9A6X ZB2 ], c_nationkey: [24], c_phone: [34-999-404-9770],  
c_acctbal: [582.88], c_mktsegment: [FURNITURE], c_comment: [fully even deposits boost. quickly ironic requests could have to are according to the  
final accounts. final]  
.c_custkey: [62707], c_name: [Customer#000062707], c_address: [bRbtSTIXhh], c_nationkey: [24], c_phone: [34-999-473-9448], c_acctbal: [7429.32], c_  
mktsegment: [FURNITURE], c_comment: [zzle slyly after the slyly ironic pinto beans. bold, fina]  
.c_custkey: [41774], c_name: [Customer#000041774], c_address: [DpHfJFKVkr4LQeif9t5SSKzDUa4G3asKeya5M7dz], c_nationkey: [24], c_phone: [34-999-535-1  
203], c_acctbal: [7118.29], c_mktsegment: [FURNITURE], c_comment: [c requests sleep. quickly silent requests wake fluffily. furiously close package  
s nag furiously across the caref]  
.c_custkey: [94308], c_name: [Customer#000094308], c_address: [5aE9PLelngeVg8v6KCg0yR,NhDT5uC4], c_nationkey: [24], c_phone: [34-999-618-6653], c_a  
cctbal: [8636.13], c_mktsegment: [FURNITURE], c_comment: [g, even deposits. carefully unus]  
.c_custkey: [102414], c_name: [Customer#000102414], c_address: [Xzg9eiWbsxiQW,LUCyUAcPoOENdLUQt], c_nationkey: [24], c_phone: [34-999-618-6881], c_  
acctbal: [9614.62], c_mktsegment: [MACHINERY], c_comment: [usly final packages. carefully even packages by the permanently special gifts ne]
```

```
scanned 150000 recs  
Calling cleanup!
```

```
[naman@0xblad3] - (~/.git/database-from-scratch/src/submit) - [git://mile2 x] -
```

```
[0] 0:zsh- 1:zsh*
```

```
CPU:8.7%
```


Test case 3

```
> ./test.out < tmp/test3.txt

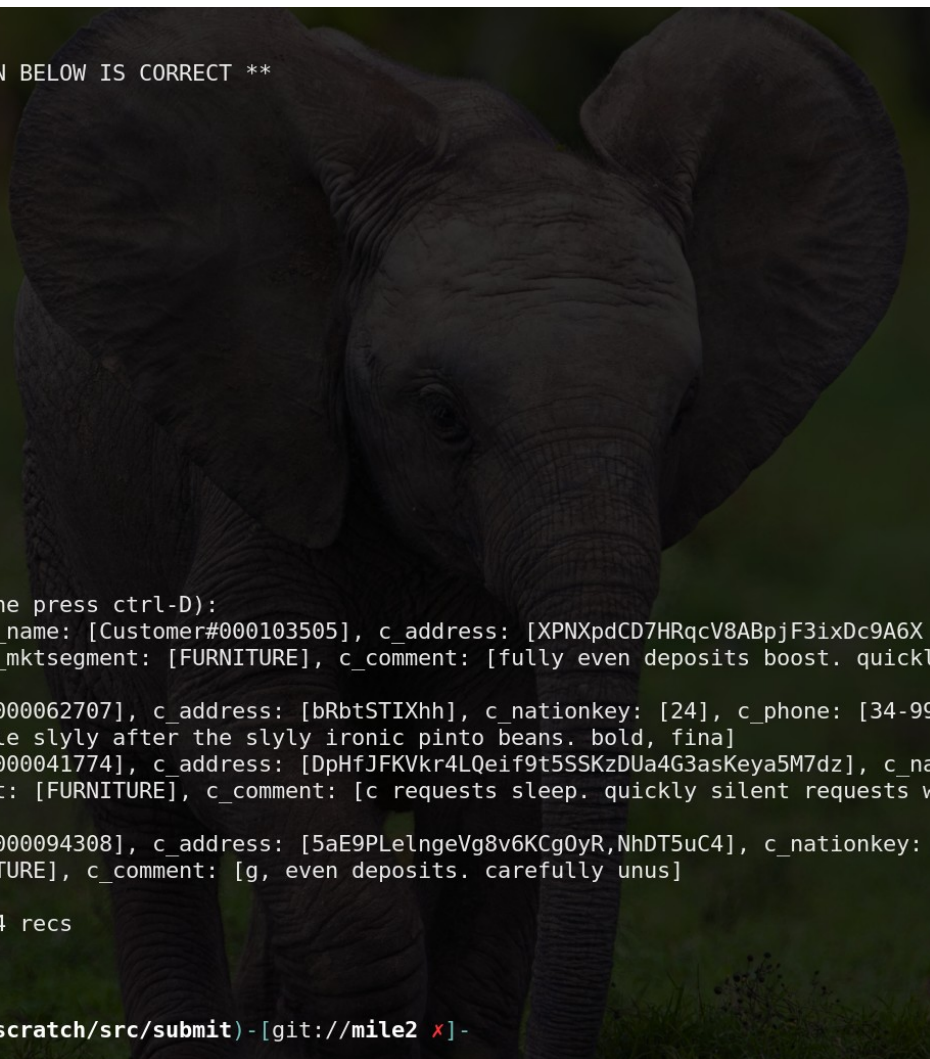
** IMPORTANT: MAKE SURE THE INFORMATION BELOW IS CORRECT **
catalog location:      catalog
tpch files dir:        tpch-dbgen/
heap files dir:        bin/

select test option:
  1. create sorted dbfile
  2. scan a dbfile
  3. run some query

select table:
  1. nation
  2. region
  3. customer
  4. part
  5. partsupp
  6. supplier
  7. orders
  8. lineitem
  Enter CNF predicate (when done press ctrl-D):
    c_custkey: [103505], c_name: [Customer#000103505], c_address: [XPNXpdCD7HRqcV8ABpjF3ixDc9A6X ZB2 ], c_nationkey: [24], c_phone: [34-999-404-9770], c_acctbal: [582.88], c_mktsegment: [FURNITURE], c_comment: [fully even deposits boost. quickly ironic requests could have to are according to the final accounts. final]
    c_custkey: [62707], c_name: [Customer#000062707], c_address: [bRbtSTIXhh], c_nationkey: [24], c_phone: [34-999-473-9448], c_acctbal: [7429.32], c_mktsegment: [FURNITURE], c_comment: [zzle slyly after the slyly ironic pinto beans. bold, fina]
    c_custkey: [41774], c_name: [Customer#000041774], c_address: [DpHfJFKVkr4LQeif9t5SSKzDUa4G3asKeya5M7dz], c_nationkey: [24], c_phone: [34-999-535-1203], c_acctbal: [7118.29], c_mktsegment: [FURNITURE], c_comment: [c requests sleep. quickly silent requests wake fluffily. furiously close packages nag furiously across the caref]
    c_custkey: [94308], c_name: [Customer#000094308], c_address: [5aE9PLelngeVg8v6KCg0yR,NhDT5uC4], c_nationkey: [24], c_phone: [34-999-618-6653], c_acctbal: [8636.13], c_mktsegment: [FURNITURE], c_comment: [g, even deposits. carefully unus]

  query over bin/customer.bin returned 4 recs
  Calling cleanup!

[naman@0xblad3] - (~/.git/database-from-scratch/src/submit) - [git://mile2 x]-
[0] 0:./test.out- 1:zsh*
```



GTest

```
[naman@0xb1ad3]-(~/git/database-from-scratch/src/submit)-[git://mile2 x]-
[> ./gtest.out

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

[=====] Running 8 tests from 4 test cases.
[-----] Global test environment set-up.
[-----] 3 tests from DBFILETEST
[ RUN      ] DBFILETEST.DBCreate
tpch file will be loaded from tpch-dbgen/customer.tbl
EOF!
[      OK  ] DBFILETEST.DBCreate (327 ms)
[ RUN      ] DBFILETEST.DBOpen
[      OK  ] DBFILETEST.DBOpen (0 ms)
[ RUN      ] DBFILETEST.DBClose
[      OK  ] DBFILETEST.DBClose (0 ms)
[-----] 3 tests from DBFILETEST (327 ms total)

[-----] 1 test from RUNGEN_DBSETUPTEST
[ RUN      ] RUNGEN_DBSETUPTEST.DBSetup
[      OK  ] RUNGEN_DBSETUPTEST.DBSetup (0 ms)
[-----] 1 test from RUNGEN_DBSETUPTEST (0 ms total)

[-----] 3 tests from TOURNAMENTTEST
[ RUN      ] TOURNAMENTTEST.Playnum
[      OK  ] TOURNAMENTTEST.Playnum (0 ms)
[ RUN      ] TOURNAMENTTEST.Promote
[      OK  ] TOURNAMENTTEST.Promote (0 ms)
[ RUN      ] TOURNAMENTTEST.InvalidNumberOfPlayer
[      OK  ] TOURNAMENTTEST.InvalidNumberOfPlayer (1 ms)
[-----] 3 tests from TOURNAMENTTEST (1 ms total)

[-----] 1 test from SORTEDFILETEST
[ RUN      ] SORTEDFILETEST.GetNext
[      OK  ] SORTEDFILETEST.GetNext (0 ms)
[-----] 1 test from SORTEDFILETEST (1 ms total)

[-----] Global test environment tear-down
[=====] 8 tests from 4 test cases ran. (329 ms total)
[ PASSED  ] 8 tests.
Calling cleanup!
[naman@0xb1ad3]-(~/git/database-from-scratch/src/submit)-[git://mile2 x]-
[> ]
```