Project Report 2

Part 2

Implementing a Sorted File

Arpan Banerjee

UFID: 9359-9083

[arpanbanerjee@ufl.edu](mailto:arpanbanerjee@ufl.edu)

Krutantak Patil

UFID: 5615-6343

Krutantakb.patil@ufl.edu

***1) Steps to compile and run - code, tests and gtests***

*(*Assuming .bin files would be provided to us in dbfiles folder)

1. **make main** – Command to build the main program.
2. **./main** – Command to run the main executable.
3. **make test.out** – Command to build the tests.
4. **./test.out** – Command to run the tests.
5. **make dBFileTest.out** – Command to build the gtests.
6. **./gtestScript.sh** – Command to run the gtests. (gtest folder contains data required for running all 12 test cases written in DBFileTest.cc file)

***2) Sorted file implementation***

**Data members**

1. **int runlen** – Used to store length of each run.
2. **File runFile** – Used to store records and pages in the file.
3. **OrderMaker sortOrder** – Pointer to page holding recent record additions to the file.
4. **Page p** – Used to temporarily store records before adding into the input pipe and also while getting records using GetNext.
5. **mode fileMode** – Stores the current mode of the file as an enum. Has values either read or write.

**Member functions**

1. **ToggleMode** (mode newMode)
   1. *Used to switch the mode of the file to read or write.*
   2. *If switching to read mode: Two way merging of the BigQ (diff file) and existing sorted file occurs.*
   3. *If switching to write mode: We clear the page object to enable writing into it.*

1. **producerWorker** (void\* arg)

* 1. Worker routine used to load data into input pipe for BigQ -
  2. Used in the Load() function to read all records from given file and insert into input pipe.
  3. Also used for merging, wherein all the records from the temporary page are inserted into the pipe.

1. **consumerWorker** (void \*arg)
   1. It is a worker routine which reads the data from the ouput pipe and inserts into pages in a file.
2. **TwoPassMergeing** ()
   1. This function first creates threads for the producer and consumer workers.
   2. It then initializes a BigQ instance which sorts in runs and merges them, releasing records in the output pipe.
   3. The threads are then joined completing the merge process.
3. **Create** (char \*filePath, fileTypeEnum fileType, void \*startup)
   1. This creates a new sorted file using the runLength and OrderMaker provided in the startup argument.
   2. It then sets the file mode to write.
4. **Load** (Schema &fileSchema, char \*loadPath)
   1. This function loads data from the given loadPath argument and uses the worker threads as described before.
5. **Open** (char \*filePath)
   1. This function opens an existing sorted file.
   2. The file mode is set to read by default.
   3. The seek is moved to the start using MoveFirst.
6. **MoveFirst** ()
   1. This first toggles the file mode to read and then sets the read page as the first page.
7. **Add** (Record &rec)
   1. This first switches the mode to write and then adds the record to the temporary page.
8. **GetNext** (Record &fetchMe)
   1. This function first toggles the mode to read.
   2. Then the page is loaded using GetPage() and records are extracted using GetFirst().
9. **GetNext** (Record &fetchMe, CNF &cnf, Record &literals)
   1. This is similar to GetNext without a CNF. The records are compared against the CNF along with the optimizations mentioned in the document.

***3) ./test.out results for 1GB data***

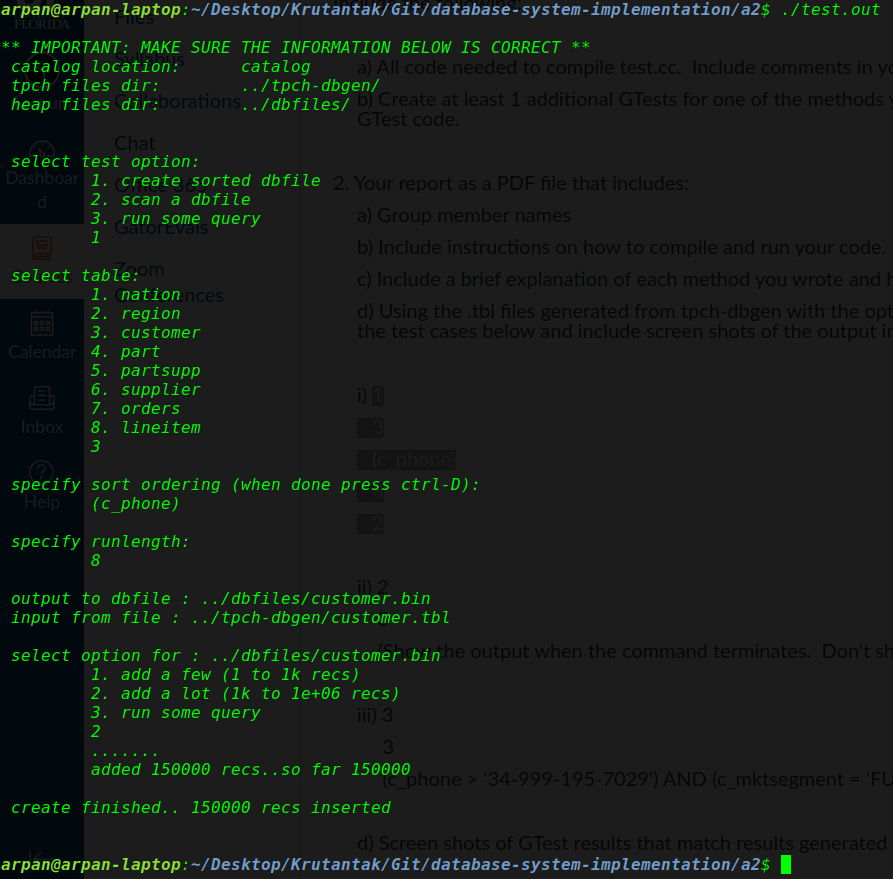
1. 1

3

(c\_phone)

8

2



1. 2

3



1. 3

3

(c\_phone > '34-999-195-7029') AND (c\_mktsegment = 'FURNITURE')



***5) gTest results:***

1. **TEST(DBFileTest, Sorted\_Create\_Equal) –** 
   1. A sorted Dbfile is created using nation.tbl. This checks the Create function.
2. **TEST(DBFileTest, Sorted\_Create\_NotEqual) –** 
   1. This function expects an exit when no table filename is provided to Create.
3. **TEST(DBFileTest, Sorted\_Open\_Equal) –** 
   1. This function tests the Open function by opening region.bin (must be created beforehand).
4. **TEST(DBFileTest, Sorted\_Close\_Empty\_File) –** 
   1. This function tests the Close function by creating a new sorted DBFile and then closing it.
5. **TEST(DBFileTest, Sorted\_Close\_Nations\_DBFile) –** 
   1. This function creates a sorted DBFile, loads data from nation.tbl and then closes it.

