

Report Assignment 1: Implementing a Heap File

Group Members:

1. Smrati Pandey, UFID: 43239459
2. Wins Goyal, UFID: 73571559

Instructions for Execution/ Code compilation and test run:

- a. Extract the folder.
- b. Navigate to the extracted folder.
- c. Execute the following commands.
Please note that the tpch files should be generated using the dbgen program before execution.
 - I. **\$ make clean**
 - II. **\$ make test.out**
 - III. **\$./test.out**
- d. Following three options will be prompted
 1. load (read the file and outputs a heap DBFile)
 2. scan (read records from an existing heap DBFile)
 3. scan & filter (read records and filter using a CNF predicate)

Brief explanation of the methods implemented and how it works.

We have used one buffer for maintaining the read or write page.

The following variables are used: to maintain various states:

- isFileOpen: To check if the File is open
 - isPageDirty: To switch between concurrent read and writes
 - isRead : To check if the page is in read mode
- a. DBFile::Create:
 - This function uses the Open method to create a binary database file. If the file is successfully created then we update the isFileOpen is set to 1 and return 1, otherwise 0 is returned. We set the variable isRead as false and isPageDirty as false.
 - b. DBFile::Load:
 - Every record from the given text file is extracted using SuckNextRecord function present within the Record class until the end of file is reached. We are reading the file from the path provided. Then it, iteratively calls the Add record to read each and every record and load it to the text file.
 - c. DBFile::Open:
 - We check if the file is already open otherwise we open the file and instantiate the buffer page in the DBFile. We assign the variable name isFileOpen to 1 otherwise we return zero to the function.

d. DBFile::MoveFirst:

- This function uses the pointer to move to the first record pointer in the file. This can be used to move the pointer to the record retrievals when needed.

e. DBFile::Close:

- The close function would return 0 if the DBFile is not open. When we are trying to close the file then we write the buffer data back to the file.

f. DBFile::Add:

- This function is used to fetch the page we are currently writing on and will further append the record on the page and write it to the memory. The page will be appended to the file.

g. DBFile::GetNext (Record &fetchme):

- Before reading the data we will check if the buffer contains a record and write the data in the memory and then start reading the data from the file and scan the records. Here, we will check all the pages present in the file and scan and fetch those records.

h. DBFile::GetNext (Record &fetchme, CNF &applyMe, Record &literal):

- This function makes an internal call to getNext and checks the CNF predicate that has been entered in the system.

Test Cases and Output Screenshots

a. q1 (*r_name* = 'EUROPE')

Using the 10MB file

```
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
  2
```

Filter with CNF for : region

Enter CNF predicate (when done press ctrl-D):

(*r_name* = 'EUROPE')

```
r_regionkey: [3], r_name: [EUROPE], r_comment: [ly final courts cajole furiously final excuse]
selected 1 recs
```

Using the 1GB file

```
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
  2

Filter with CNF for : region
Enter CNF predicate (when done press ctrl-D):
  (r_name = 'EUROPE')
r_regionkey: [3], r_name: [EUROPE], r_comment: [ly final courts cajole furiously final excuse]
selected 1 recs
```

b. q2 (r_name < 'middle east') AND (r_regionkey > 1)

Using 10MB File

```
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
  2

Filter with CNF for : region
Enter CNF predicate (when done press ctrl-D):
  (r_name < 'middle east') AND (r_regionkey > 1)
r_regionkey: [2], r_name: [ASIA], r_comment: [ges. thinly even pinto beans ca]
r_regionkey: [3], r_name: [EUROPE], r_comment: [ly final courts cajole furiously final excuse]
r_regionkey: [4], r_name: [MIDDLE EAST], r_comment: [uickly special accounts cajole carefully blithely close requests. carefully final asymptotes haggle furiousl]
selected 3 recs
```

Using 1GB File

```
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
  2

Filter with CNF for : region
Enter CNF predicate (when done press ctrl-D):
  (r_name < 'middle east') AND
(r_regionkey > 1)
r_regionkey: [2], r_name: [ASIA], r_comment: [ges. thinly even pinto beans ca]
r_regionkey: [3], r_name: [EUROPE], r_comment: [ly final courts cajole furiously final excuse]
r_regionkey: [4], r_name: [MIDDLE EAST], r_comment: [uickly special accounts cajole carefully blithely close requests. carefully final asymptotes haggle furiously]
selected 3 recs
```

c. q3 (n_regionkey = 3) AND (n_nationkey > 10) AND (n_name > 'japan')

Use 10MB File

```
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
  1

Filter with CNF for : nation
Enter CNF predicate (when done press ctrl-D):
  (n_regionkey = 3) AND
(n_nationkey > 10) AND
(n_name > 'japan')
selected 0 recs
```

Use 1GB File

```
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
  1

Filter with CNF for : nation
Enter CNF predicate (when done press ctrl-D):
  (n_regionkey = 3) AND
(n_nationkey > 10) AND
(n_name > 'japan')
selected 0 recs
```

d. q11 (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)

Use 10MB file

```
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_shipdate > '1994-01-01') AND
(l_shipdate < '1994-01-07') AND
(l_discount > 0.05) AND
(l_discount < 0.06) AND
(l_quantity = 4.0)
selected 0 recs
```

Use 1GB data

```
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):
  (_shipdate > '1994-01-01') AND
(_shipdate < '1994-01-07') AND
(_discount > 0.05) AND
(_discount < 0.06) AND
(_quantity = 4.0)
selected 0 recs
```

e. q12 (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)

Use 10MB

```
te: [1994-03-30], l_shipinstruct: [DELIVER IN PERSON], l_shipmode: [AIR], l_comment: [refully final theodoli]
l_orderkey: [416], l_partkey: [1742], l_suppkey: [85], l_linenumber: [3], l_quantity: [25], l_extendedprice: [41893.5], l_discount: [0.07], l_tax: [0.01], l_returnflag: [R], l_linestatus: [F], l_shipdate: [1993-10-16], l_commitdate: [1993-12-03], l_receiptdate: [1993-10-29], l_shipinstruct: [NONE], l_shipmode: [AIR], l_comment: [ses boost after the bold requests.]
l_orderkey: [450], l_partkey: [781], l_suppkey: [82], l_linenumber: [5], l_quantity: [2], l_extendedprice: [3363.56], l_discount: [0.09], l_tax: [0], l_returnflag: [A], l_linestatus: [F], l_shipdate: [1995-03-11], l_commitdate: [1995-05-21], l_receiptdate: [1995-03-16], l_shipinstruct: [COLLECT C00], l_shipmode: [AIR], l_comment: [y even pinto beans; qui]
l_orderkey: [481], l_partkey: [207], l_suppkey: [8], l_linenumber: [2], l_quantity: [19], l_extendedprice: [21036.8], l_discount: [0.08], l_tax: [0.01], l_returnflag: [R], l_linestatus: [F], l_shipdate: [1993-01-09], l_commitdate: [1992-11-27], l_receiptdate: [1993-01-14], l_shipinstruct: [TAKE BACK RETURN], l_shipmode: [AIR], l_comment: [p blithely after t]
l_orderkey: [481], l_partkey: [1120], l_suppkey: [93], l_linenumber: [5], l_quantity: [31], l_extendedprice: [31654.7], l_discount: [0.05], l_tax: [0.01], l_returnflag: [A], l_linestatus: [F], l_shipdate: [1993-01-15], l_commitdate: [1992-12-31], l_receiptdate: [1993-01-21], l_shipinstruct: [DELIVER IN PERSON], l_shipmode: [AIR], l_comment: [usly final packages believe. quick]
l_orderkey: [548], l_partkey: [568], l_suppkey: [59], l_linenumber: [4], l_quantity: [21], l_extendedprice: [30839.8], l_discount: [0.08], l_tax: [0.03], l_returnflag: [A], l_linestatus: [F], l_shipdate: [1994-10-27], l_commitdate: [1994-12-04], l_receiptdate: [1993-04-07], l_shipinstruct: [DELIVER IN PERSON], l_shipmode: [AIR], l_comment: [ engage quickly. regular theo]
l_orderkey: [549], l_partkey: [653], l_suppkey: [54], l_linenumber: [3], l_quantity: [36], l_extendedprice: [55931.4], l_discount: [0.04], l_tax: [0.04], l_returnflag: [R], l_linestatus: [F], l_shipdate: [1992-09-11], l_commitdate: [1992-10-11], l_receiptdate: [1992-09-12], l_shipinstruct: [DELIVER IN PERSON], l_shipmode: [AIR], l_comment: [ts against the ironic, even theodolites eng]
l_orderkey: [577], l_partkey: [259], l_suppkey: [14], l_linenumber: [1], l_quantity: [25], l_extendedprice: [20901.2], l_discount: [0.06], l_tax: [0.01], l_returnflag: [A], l_linestatus: [F], l_shipdate: [1995-04-09], l_commitdate: [1995-02-20], l_receiptdate: [1995-05-09], l_shipinstruct: [TAKE BACK RETURN], l_shipmode: [AIR], l_comment: [ve slyly of the frets. careful]
l_orderkey: [677], l_partkey: [1497], l_suppkey: [15], l_linenumber: [5], l_quantity: [25], l_extendedprice: [34962.2], l_discount: [0], l_tax: [0.05], l_returnflag: [A], l_linestatus: [F], l_shipdate: [1994-03-12], l_commitdate: [1994-02-02], l_receiptdate: [1994-03-28], l_shipinstruct: [DELIVER IN PERSON], l_shipmode: [AIR], l_comment: [ packages integrate blithely]
l_orderkey: [678], l_partkey: [975], l_suppkey: [76], l_linenumber: [5], l_quantity: [16], l_extendedprice: [30015.5], l_discount: [0.06], l_tax: [0.04], l_returnflag: [R], l_linestatus: [F], l_shipdate: [1993-03-09], l_commitdate: [1993-04-18], l_receiptdate: [1993-04-07], l_shipinstruct: [NONE], l_shipmode: [AIR], l_comment: [ about the ]
l_orderkey: [738], l_partkey: [1871], l_suppkey: [72], l_linenumber: [2], l_quantity: [4], l_extendedprice: [7091.48], l_discount: [0], l_tax: [0.03], l_returnflag: [A], l_linestatus: [F], l_shipdate: [1993-06-20], l_commitdate: [1993-04-08], l_receiptdate: [1993-07-09], l_shipinstruct: [NONE], l_shipmode: [AIR], l_comment: [ar packages. fluffily bo]
l_orderkey: [738], l_partkey: [1749], l_suppkey: [34], l_linenumber: [5], l_quantity: [30], l_extendedprice: [49522.2], l_discount: [0.02], l_tax: [0], l_returnflag: [A], l_linestatus: [F], l_shipdate: [1993-06-12], l_commitdate: [1993-05-29], l_receiptdate: [1993-06-25], l_shipinstruct: [NONE], l_shipmode: [AIR], l_comment: [ecial instructions haggle blithely regula]
l_orderkey: [742], l_partkey: [1919], l_suppkey: [64], l_linenumber: [4], l_quantity: [16], l_extendedprice: [29134.6], l_discount: [0.01], l_tax: [0.05], l_returnflag: [A], l_linestatus: [F], l_shipdate: [1995-01-15], l_commitdate: [1995-02-25], l_receiptdate: [1995-01-24], l_shipinstruct: [COLLECT C00], l_shipmode: [AIR], l_comment: [eodolites haggle carefully regul]
l_orderkey: [769], l_partkey: [1751], l_suppkey: [94], l_linenumber: [1], l_quantity: [36], l_extendedprice: [59499], l_discount: [0.02], l_tax: [0.02], l_returnflag: [A], l_linestatus: [F], l_shipdate: [1993-10-01], l_commitdate: [1993-08-07], l_receiptdate: [1993-10-15], l_shipinstruct: [NONE], l_shipmode: [AIR], l_comment: [es. furiously iro]
l_orderkey: [772], l_partkey: [1799], l_suppkey: [26], l_linenumber: [4], l_quantity: [10], l_extendedprice: [17007.9], l_discount: [0.08], l_tax: [0.02], l_returnflag: [A], l_linestatus: [F], l_shipdate: [1993-05-17], l_commitdate: [1993-06-09], l_receiptdate: [1993-05-29], l_shipinstruct: [COLLECT C00], l_shipmode: [AIR], l_comment: [o the furiously final deposits. furi]
l_orderkey: [801], l_partkey: [947], l_suppkey: [16], l_linenumber: [2], l_quantity: [21], l_extendedprice: [38806.7], l_discount: [0.05], l_tax: [0.02], l_returnflag: [A], l_linestatus: [F], l_shipdate: [1992-03-14], l_commitdate: [1992-04-01], l_receiptdate: [1992-04-05], l_shipinstruct: [COLLECT C00], l_shipmode: [AIR], l_comment: [wake silently furiously idle deposits. ]
l_orderkey: [802], l_partkey: [1328], l_suppkey: [29], l_linenumber: [2], l_quantity: [34], l_extendedprice: [41796.9], l_discount: [0.08], l_tax: [0.06], l_returnflag: [A], l_linestatus: [F], l_shipdate: [1995-03-01], l_commitdate: [1995-03-15], l_receiptdate: [1995-03-12], l_shipinstruct: [COLLECT C00], l_shipmode: [AIR], l_comment: [instructions cajole carefully. quiet]
l_orderkey: [866], l_partkey: [1358], l_suppkey: [59], l_linenumber: [1], l_quantity: [5], l_extendedprice: [6296.75], l_discount: [0.08], l_tax: [0], l_returnflag: [R], l_linestatus: [F], l_shipdate: [1993-01-22], l_commitdate: [1993-01-14], l_receiptdate: [1993-02-07], l_shipinstruct: [TAKE BACK RETURN], l_shipmode: [AIR], l_comment: [tegrate fluffily. carefully f]
l_orderkey: [928], l_partkey: [104], l_suppkey: [31], l_linenumber: [7], l_quantity: [11], l_extendedprice: [11045.1], l_discount: [0], l_tax: [0.01], l_returnflag: [A], l_linestatus: [F], l_shipdate: [1995-04-29], l_commitdate: [1995-04-16], l_receiptdate: [1995-04-30], l_shipinstruct: [NONE], l_shipmode: [AIR], l_comment: [longside of]
l_orderkey: [930], l_partkey: [173], l_suppkey: [52], l_linenumber: [2], l_quantity: [47], l_extendedprice: [50439], l_discount: [0.08], l_tax: [0], l_returnflag: [A], l_linestatus: [F], l_shipdate: [1995-03-20], l_commitdate: [1995-02-04], l_receiptdate: [1995-04-04], l_shipinstruct: [DELIVER IN PERSON], l_shipmode: [AIR], l_comment: [ackages. fluffily e]
l_orderkey: [933], l_partkey: [992], l_suppkey: [27], l_linenumber: [3], l_quantity: [26], l_extendedprice: [49217.7], l_discount: [0.05], l_tax: [0], l_returnflag: [A], l_linestatus: [F], l_shipdate: [1992-11-09], l_commitdate: [1992-11-03], l_receiptdate: [1992-11-16], l_shipinstruct: [DELIVER IN PERSON], l_shipmode: [AIR], l_comment: [ the deposits affix slyly after t]
l_orderkey: [960], l_partkey: [1070], l_suppkey: [6], l_linenumber: [1], l_quantity: [1], l_extendedprice: [971.07], l_discount: [0.07], l_tax: [0], l_returnflag: [A], l_linestatus: [F], l_shipdate: [1994-12-24], l_commitdate: [1994-10-26], l_receiptdate: [1995-01-20], l_shipinstruct: [DELIVER IN PERSON], l_shipmode: [AIR], l_comment: [y ironic packages. quickly even ]
selected 35 recs
```

Use 1GB File

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)

l_orderkey: [130], l_partkey: [1739], l_suppkey: [4240], l_linenum: [2], l_quantity: [48], l_extendedprice: [78755], l_discount: [0.03], l_tax: [0.02], l_returnflag: [R], l_linestatus: [F], l_shipdate: [1992-07-01], l_commitdate: [1992-07-12], l_receiptdate: [1992-07-24], l_shipinstruct: [NONE], l_shipmode: [AIR], l_comment: [lithely alongside of the regu]

l_orderkey: [194], l_partkey: [2594], l_suppkey: [5095], l_linenum: [1], l_quantity: [17], l_extendedprice: [25442], l_discount: [0.05], l_tax: [0.04], l_returnflag: [R], l_linestatus: [F], l_shipdate: [1992-05-24], l_commitdate: [1992-05-22], l_receiptdate: [1992-05-30], l_shipinstruct: [COLLECT COD], l_shipmode: [AIR], l_comment: [regular deposi]

selected 2 recs

Instructions to run Gtest

1) make clean

2) make myGtest.out

3) ./myGtest.out

```
smrati@smrati-XPS-15-7590:~/Downloads/p1/P1$ ./myGtest.out
```

```
[=====] Running 4 tests from 1 test case.
```

```
[-----] Global test environment set-up.
```

```
[-----] 4 tests from DBFile
```

```
[ RUN      ] DBFile.TestCreateSuccess
```

```
Successfully Created the File
```

```
[ OK      ] DBFile.TestCreateSuccess (1 ms)
```

```
[ RUN      ] DBFile.FailureOpen
```

```
The File is already Open
```

```
Failed to Open File
```

```
[ OK      ] DBFile.FailureOpen (0 ms)
```

```
[ RUN      ] DBFile.TestClose
```

```
Successfully Closed the File
```

```
File is not Open
```

```
File is already closed
```

```
[ OK      ] DBFile.TestClose (0 ms)
```

```
[ RUN      ] DBFile.SuccessOpen
```

```
File Opened Successfully
```

```
[ OK      ] DBFile.SuccessOpen (0 ms)
```

```
[-----] 4 tests from DBFile (1 ms total)
```

```
[-----] Global test environment tear-down
```

```
[=====] 4 tests from 1 test case ran. (1 ms total)
```

```
[ PASSED ] 4 tests.
```

As mentioned, we have made the test cases to test DBFile::Create, DBFile::Open and DBFile::Close methods.

- 1) TestCreateSuccess: It creates the file with "lineitem.bin", and check the status of creation i.e. after the Create should return 1.
- 2) FailureOpen: It tests DBFile::Open method. This test case tries to open "temporary.bin" and expects a value of 0 as file is not present inside the directory.
- 3) TestClose: It tests DBFile::Close method. It first creates the file and then tries to close the file. Close should return 1 and close the file if file is already open, and return 0 if file is not open.
- 4) SuccessOpen: It Success scenario of DBFile::Open method. It creates one file and then close that file, after that it open the file using DBFile open method and expects it to return success response i.e. 1.