

# 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 4\_2. The query optimization 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 TPCB 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: ‘yyfunclinen0’ 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:

Pipe id for input from database file is -2 and Pipe id for final output is -1

### 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\_p42.zip has one sub-directory, src/.

```
$> unzip NamanArora_NikunjSarda_p42.zip
```

```
$> cd proj42
```

Now to build:

```
$> make
```

The above command will create a42.out

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 qp\_tree.cc
- ➔ Added qp\_tree\_helper.cc

### qp\_tree.cc:

- ➔ query :: query(struct FuncOperator \*finalFunction, struct TableList \*tables, struct AndList \*boolean, struct NameList \*groupingAtts, struct NameList \*attsToSelect, int distinctAtts, int distinctFunc): constructor to structure query, initializing the variable to initial value
- ➔ query :: ~query(): Destructor to structure query
- ➔ operation :: operation(): constructor to structure operation, initializing the variable to initial value
- ➔ operation :: operation(int flag): constructor to structure operation, initializing the variable to initial value and type (struct variable) to flag
- ➔ operation :: operation(struct AndList \*a\_list, Qptree \*ref): constructor to structure operation, initializing the variable to provided values
- ➔ operation :: ~operation(): destructor to structure operation
- ➔ void operation :: print(): method to print process information
- ➔ bool sel\_op\_comp :: operator()(operation \*l, operation \*r): method to compare cost of left and right operation and return boolean value accordingly
  - ➔ bool join\_op\_comp :: operator()(operation \*l, operation \*r): method to compare cost of left and right operation and return boolean value accordingly
- ➔ tableInfo :: tableInfo(): constructor to structure tableInfo
- ➔ tableInfo :: ~tableInfo(): destructor to structure tableInfo
- ➔ struct operation \*tableInfo :: dispense\_select(): method to evaluate cost for select
- ➔ Qptree :: Qptree(char \*stat\_fname, char \*catalog\_file): constructor to class Qptree to initialize class variable
- ➔ Qptree :: ~Qptree(): destructor to class Qptree
- ➔ void Qptree :: process(struct query \*q): method to construct the expense tree
- ➔ void print\_in\_order(struct operation \*tree): method to print the expense tree

### qp\_tree\_helper.cc:

- ➔ void Qptree :: get\_attr(char \*att\_name, pair<string, unordered\_map<string, tableInfo> :: iterator> &p): method to get the attribute via name passed to the function and stored it in pair object
- ➔ void Qptree :: process(struct TableList \*tables): method to process info for given tables list

- ➔ `void Qptree :: process(struct operation *op, struct AndList *a_list, struct OrList *o_list, char **rels, int *curr_indx):` method to process “or” operation
- ➔ `void Qptree :: process(struct AndList *a_list):` method to process “and” operator starting from right end
- ➔ `int Qptree :: dispense_pipe():` method to return pipe count
- ➔ `struct operation *Qptree :: dispense_join(struct operation *j_op, int indx, vector<operation *> &j_vec, stack<operation *> &j_stk):` method to get least cost join
- ➔ `void Qptree :: process_join(struct operation *j_op, vector<operation *> &j_vec, stack<operation *> &j_stk):` method to process joins according to there expense

Outputs:

## output42.txt

```
src 0 exec 1 Makefile 2 git 3 zsh
1 c1
2 Overwriting file tmp/statistics.txt
3 Enter the query:
4 Printing the tree in order!
5 *****
6 SELECT FILE
7 CNF:
8 ( Att 1 from left record = Att 0 from literal record (String))
9 Input pipe ID: -2
10 Output pipe ID: 0
11 Output Schema:
12 n_nationkey Int
13 n_name String
14 n_regionkey Int
15 n_comment String
16 *****
17 *****
18 PROJECT:
19 Output Attributes:
20 n.n_nationkey
21
22 Input pipe ID: 0
23 Output pipe ID: -1
24 Input Schema:
25 n_nationkey Int
26 n_name String
27 n_regionkey Int
28 n_comment String
29 *****
30 *****
31 TC2
32 Overwriting file tmp/statistics.txt
33 Enter the query:
34 Printing the tree in order!
35 *****
36 SELECT FILE
37 CNF:
38 ( Att 0 from left record > Att 0 from literal record (Int))
39 Input pipe ID: -2
40 Output pipe ID: 0
41 Output Schema:
42 n_nationkey Int
43 n_name String
44 n_regionkey Int
45 n_comment String
46 *****
NORMAL tmp/output42.txt text utf-8[unix] 999 words 0% 1/431 : 1 [3]trailing
"tmp/output42.txt" 431L, 7451C
```

## output42.txt

```
src 0 exec 1 Makefile 2 git 3 zsh
47 *****
48 JOIN OPERATION:
49 CNF:
50 ( Att 2 from left record = Att 0 from right record (Int))
51 Input pipe ID: 0, 1
52 Output pipe ID: 2
53 Output Schema:
54 n_nationkey Int
55 n_name String
56 n_regionkey Int
57 n_comment String
58 r_regionkey Int
59 r_name String
60 r_comment String
61 *****
62 *****
63 SELECT FILE
64 CNF:
65 Input pipe ID: -2
66 Output pipe ID: 1
67 Output Schema:
68 r_regionkey Int
69 r_name String
70 r_comment String
71 *****
72 *****
73 PROJECT:
74 Output Attributes:
75 n.n_name
76
77 Input pipe ID: 2
78 Output pipe ID: -1
79 Input Schema:
80 n_nationkey Int
81 n_name String
82 n_regionkey Int
83 n_comment String
84 r_regionkey Int
85 r_name String
86 r_comment String
87 *****
88 *****
89 TC3
90 Overwriting file tmp/statistics.txt
91 Enter the query:
92 Printing the tree in order!
NORMAL tmp/output42.txt
text utf-8[unix] 999 words 11% 48/431 : 1 [3]trailing
```



## output42.txt

```
src 0 exec 1 Makefile 2 git 3 zsh
90 Overwriting file tmp/statistics.txt
91 Enter the query:
92 Printing the tree in order!
93 *****
94 SELECT FILE
95 CNF:
96 ( Att 1 from left record = Att 0 from literal record (String))
97 Input pipe ID: -2
98 Output pipe ID: 0
99 Output Schema:
100 n_nationkey Int
101 n_name String
102 n_regionkey Int
103 n_comment String
104 *****
105 *****
106 JOIN OPERATION:
107 CNF:
108 ( Att 2 from left record = Att 0 from right record (Int))
109 Input pipe ID: 0, 1
110 Output pipe ID: 2
111 Output Schema:
112 n_nationkey Int
113 n_name String
114 n_regionkey Int
115 n_comment String
116 r_regionkey Int
117 r_name String
118 r_comment String
119 *****
120 *****
121 SELECT FILE
122 CNF:
123 Input pipe ID: -2
124 Output pipe ID: 1
125 Output Schema:
126 r_regionkey Int
127 r_name String
128 r_comment String
129 *****
130 *****
131 SUM
132 Function:
133 n.n_nationkey
134 Input pipe ID: 2
135 Output pipe ID: 3
NORMAL tmp/output42.txt
```

95% naman

text utf-8[unix] 999 words 31% 135/431 : 1 [3]trailing

## output42.txt

```
src 0 exec 1 Makefile 2 git 3 zsh
134 Input pipe ID: 2
135 Output pipe ID: 3
136 Output Schema:
137 n_nationkey Int
138 n_name String
139 n_regionkey Int
140 n_comment String
141 r_regionkey Int
142 r_name String
143 r_comment String
144 *****
145 *****
146 PROJECT:
147 Output Attributes:
148 n.n_nationkey,
149
150 Input pipe ID: 3
151 Output pipe ID: -1
152 Input Schema:
153 n_nationkey Int
154 n_name String
155 n_regionkey Int
156 n_comment String
157 r_regionkey Int
158 r_name String
159 r_comment String
160 *****
161 *****
162 TC4
163 Overwriting file tmp/statistics.txt
164 Enter the query:
165 Printing the tree in order!
166 *****
167 SELECT FILE
168 CNF:
169 ( Att 1 from left record = Att 0 from literal record (String))
170 Input pipe ID: -2
171 Output pipe ID: 0
172 Output Schema:
173 n_nationkey Int
174 n_name String
175 n_regionkey Int
176 n_comment String
177 *****
178 *****
179 JOIN OPERATION:
NORMAL tmp/output42.txt
text utf-8[unix] 999 words 41% 179/431 : 1 [3]trailing
```



## output42.txt

```
src 0 exec 1 Makefile 2 git 3 zsh
178 *****
179 JOIN OPERATION:
180 CNF:
181 ( Att 2 from left record = Att 0 from right record (Int))
182 Input pipe ID: 0, 1
183 Output pipe ID: 2
184 Output Schema:
185 n_nationkey Int
186 n_name String
187 n_regionkey Int
188 n_comment String
189 r_regionkey Int
190 r_name String
191 r_comment String
192 *****
193 *****
194 SELECT FILE
195 CNF:
196 Input pipe ID: -2
197 Output pipe ID: 1
198 Output Schema:
199 r_regionkey Int
200 r_name String
201 r_comment String
202 *****
203 *****
204 GROUP BY
205 Function:
206 n.n regionkey
207 OrderMaker:
208 NumAtts = 1
209 0: 0 Int
210 Group by Schema:
211 n.n regionkey Int
212 Input pipe ID: 2
213 Output pipe ID: 3
214 Output Schema:
215 n_nationkey Int
216 n_name String
217 n_regionkey Int
218 n_comment String
219 r_regionkey Int
220 r_name String
221 r_comment String
222 *****
223 *****
NORMAL tmp/output42.txt text utf-8[unix] 999 words 51% 223/431 : 1 [3]trailing
```

## output42.txt

```
src 0 exec 1 Makefile 2 git 3 zsh
221 r comment String
222 *****
223 *****
224 SUM
225 Function:
226 n.n_regionkey
227 Input pipe ID: 3
228 Output pipe ID: 4
229 Output Schema:
230 n_nationkey Int
231 n_name String
232 n_regionkey Int
233 n_comment String
234 r_regionkey Int
235 r_name String
236 r_comment String
237 *****
238 *****
239 PROJECT:
240 Output Attributes:
241 n.n_regionkey,
242
243 Input pipe ID: 4
244 Output pipe ID: -1
245 Input Schema:
246 n_nationkey Int
247 n_name String
248 n_regionkey Int
249 n_comment String
250 r_regionkey Int
251 r_name String
252 r_comment String
253 *****
254 *****
255 TC5
256 Overwriting file tmp/statistics.txt
257 Enter the query:
258 Printing the tree in order!
259 *****
260 SELECT FILE
261 CNF:
262 ( Att 0 from left record > Att 0 from literal record (Int))
263 Input pipe ID: -2
264 Output pipe ID: 0
265 Output Schema:
266 n_nationkey Int
NORMAL tmp/output42.txt
text utf-8[unix] 999 words 61% 266/431 : 1 [3]trailing
```

## output42.txt

```
src 0 exec 1 Makefile 2 git 3 zsh
266 n_nationkey Int
267 n_name String
268 n_regionkey Int
269 n_comment String
270 *****
271 *****
272 JOIN OPERATION:
273 CNF:
274 ( Att 2 from left record = Att 0 from right record (Int))
275 Input pipe ID: 0, 1
276 Output pipe ID: 2
277 Output Schema:
278 n_nationkey Int
279 n_name String
280 n_regionkey Int
281 n_comment String
282 r_regionkey Int
283 r_name String
284 r_comment String
285 *****
286 *****
287 SELECT FILE
288 CNF:
289 Input pipe ID: -2
290 Output pipe ID: 1
291 Output Schema:
292 r_regionkey Int
293 r_name String
294 r_comment String
295 *****
296 *****
297 JOIN OPERATION:
298 CNF:
299 ( Att 0 from left record = Att 3 from right record (Int))
300 Input pipe ID: 2, 3
301 Output pipe ID: 4
302 Output Schema:
303 n_nationkey Int
304 n_name String
305 n_regionkey Int
306 n_comment String
307 c_custkey Int
308 c_name String
309 c_address String
310 c_nationkey Int
311 c_phone String
NORMAL tmp/output42.txt
text utf-8[unix] 999 words 72% 311/431 : 1 [3]trailing
```



## output42.txt

```
src 0 exec 1 Makefile 2 git 3 zsh
310 c_nationkey Int
311 c_phone String
312 c_acctbal Double
313 c_mktsegment String
314 c_comment String
315 r_regionkey Int
316 r_name String
317 r_comment String
318 *****
319 *****
320 SELECT FILE
321 CNF:
322 Input pipe ID: -2
323 Output pipe ID: 3
324 Output Schema:
325 c_custkey Int
326 c_name String
327 c_address String
328 c_nationkey Int
329 c_phone String
330 c_acctbal Double
331 c_mktsegment String
332 c_comment String
333 *****
334 *****
335 GROUP BY
336 Function:
337 n.n_nationkey+r.r_regionkey
338 OrderMaker:
339 NumAtts = 1
340 0: 0 Int
341 Group by Schema:
342 r.r_regionkey Int
343 Input pipe ID: 4
344 Output pipe ID: 5
345 Output Schema:
346 n_nationkey Int
347 n_name String
348 n_regionkey Int
349 n_comment String
350 c_custkey Int
351 c_name String
352 c_address String
353 c_nationkey Int
354 c_phone String
355 c_acctbal Double
NORMAL tmp/output42.txt
text utf-8[unix] 999 words 82% 355/431 : 1 [3]trailing
```

## output42.txt

```
src 0 exec 1 Makefile 2 git 3 zsh
355 c_acctbal Double
356 c_mktsegment String
357 c_comment String
358 r_regionkey Int
359 r_name String
360 r_comment String
361 *****
362 *****
363 DISTINCT:
364
365 Input pipe ID: 5
366 Output pipe ID: 6
367 Output Schema:
368 n_nationkey Int
369 n_name String
370 n_regionkey Int
371 n_comment String
372 c_custkey Int
373 c_name String
374 c_address String
375 c_nationkey Int
376 c_phone String
377 c_acctbal Double
378 c_mktsegment String
379 c_comment String
380 r_regionkey Int
381 r_name String
382 r_comment String
383 *****
384 *****
385 SUM
386 Function:
387 n.n.nationkey+r.r.regionkey
388 Input pipe ID: 6
389 Output pipe ID: 7
390 Output Schema:
391 n_nationkey Int
392 n_name String
393 n_regionkey Int
394 n_comment String
395 c_custkey Int
396 c_name String
397 c_address String
398 c_nationkey Int
399 c_phone String
400 c_acctbal Double
NORMAL tmp/output42.txt
text utf-8[unix] 999 words 92% 400/431 : 1 [3]trailing
```

## output42.txt

```
src 0 exec 1 Makefile 2 git 3 zsh
386 Function:
387 n.n_nationkey+r.r_regionkey
388 Input pipe ID: 6
389 Output pipe ID: 7
390 Output Schema:
391 n_nationkey Int
392 n_name String
393 n_regionkey Int
394 n_comment String
395 c_custkey Int
396 c_name String
397 c_address String
398 c_nationkey Int
399 c_phone String
400 c_acctbal Double
401 c_mktsegment String
402 c_comment String
403 r_regionkey Int
404 r_name String
405 r_comment String
406 *****
407 *****
408 PROJECT:
409 Output Attributes:
410 n.n_nationkey, , r.r_regionkey,
411
412 Input pipe ID: 7
413 Output pipe ID: -1
414 Input Schema:
415 n_nationkey Int
416 n_name String
417 n_regionkey Int
418 n_comment String
419 c_custkey Int
420 c_name String
421 c_address String
422 c_nationkey Int
423 c_phone String
424 c_acctbal Double
425 c_mktsegment String
426 c_comment String
427 r_regionkey Int
428 r_name String
429 r_comment String
430 *****
431 *****
NORMAL tmp/output42.txt text utf-8[unix] 999 words 100% 431/431 : 1 [3]trailing
```



# GTest

```
src 0 exec 1 Makefile 2 git 3 zsh
[naman@0xb1ad3] - (~/.git/database-from-scratch/src/submit) - [git://proj4_2 x]-
ls
bigq.cc          comparison.h  function.cc  lexer_func.l  parse_tree.h  record.cc      runTestCases42.sh  stat_helper.cc  tournament.cc
bigq.h           dbfile.cc   function.h   lexer.l        pipe.cc        record.h       schema.cc          statistics.cc   tournament.h
catalog          dbfile.h    gtest.cc    Makefile       pipe.h         run_gen.cc     schema.h          statistics.h    tpch-dbggen
comparison.cc    defs.h      gtest.out   parse_func.h   qp_tree.cc     run_gen.h      sorted.cc         test.cc        two_way_list.cc
comparison_engine.cc file.cc    heap.cc     parser_func.y  qp_tree.h      run_merge.cc   sorted.h          test.h         two_way_list.h
comparison_engine.h file.h     heap.h      parser.y       qp_tree_helper.cc run_merge.h    sort_helper.cc   tmp

[naman@0xb1ad3] - (~/.git/database-from-scratch/src/submit) - [git://proj4_2 x]-
./gtest.out
[=====] Running 2 tests from 1 test case.
[-----] Global test environment set-up.
[-----] 2 tests from QTREESTEST
[ RUN    ] QTREESTEST.pipecount
File [ OK ] QTREESTEST.pipecount (0 ms)
[ RUN    ] QTREESTEST.selops
[ OK     ] QTREESTEST.selops (0 ms)
[-----] 2 tests from QTREESTEST (0 ms total)

[-----] Global test environment tear-down
[=====] 2 tests from 1 test case ran. (0 ms total)
[ PASSED ] 2 tests.
[naman@0xb1ad3] - (~/.git/database-from-scratch/src/submit) - [git://proj4_2 x]-
```