Name: *Samuel Sovi*

Assignment: 5

Complete & Correct:	48 / 48
Tests:	0 / 0
Format and Comments:	6 / 6
Writeup:	6 / 6
Total Score:	60 / 60

Comments:

1. next and prev key shouldn't set the output param unless a valid key is found

Unit test output:

```
==326661== Memcheck, a memory error detector
==326661== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==326661== Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright info
==326661== Command: ./hw5_test
==326661==
[==========] Running 22 tests from 2 test suites.
[----------] Global test environment set-up.
[----------] 11 tests from BasicArrayMapTests
[ RUN      ] BasicArrayMapTests.EmptyCheck
[       OK ] BasicArrayMapTests.EmptyCheck (12 ms)
[ RUN      ] BasicArrayMapTests.InsertCheck
[       OK ] BasicArrayMapTests.InsertCheck (4 ms)
[ RUN      ] BasicArrayMapTests.RValueAccessCheck
[       OK ] BasicArrayMapTests.RValueAccessCheck (4 ms)
[ RUN      ] BasicArrayMapTests.LValueAccessCheck
[       OK ] BasicArrayMapTests.LValueAccessCheck (3 ms)
[ RUN      ] BasicArrayMapTests.ContainsCheck
[       OK ] BasicArrayMapTests.ContainsCheck (3 ms)
[ RUN      ] BasicArrayMapTests.EraseCheck
[       OK ] BasicArrayMapTests.EraseCheck (6 ms)
[ RUN      ] BasicArrayMapTests.KeyRangeCheck
[       OK ] BasicArrayMapTests.KeyRangeCheck (8 ms)
[ RUN      ] BasicArrayMapTests.SortedKeyCheck
[       OK ] BasicArrayMapTests.SortedKeyCheck (9 ms)
[ RUN      ] BasicArrayMapTests.InvalidKeyCheck
[       OK ] BasicArrayMapTests.InvalidKeyCheck (30 ms)
[ RUN      ] BasicArrayMapTests.NextKeyCheck
[       OK ] BasicArrayMapTests.NextKeyCheck (7 ms)
[ RUN      ] BasicArrayMapTests.PrevKeyCheck
[       OK ] BasicArrayMapTests.PrevKeyCheck (7 ms)
[----------] 11 tests from BasicArrayMapTests (107 ms total)

[----------] 11 tests from BasicLinkedMapTests
[ RUN      ] BasicLinkedMapTests.EmptyCheck
[       OK ] BasicLinkedMapTests.EmptyCheck (2 ms)
[ RUN      ] BasicLinkedMapTests.InsertCheck
[       OK ] BasicLinkedMapTests.InsertCheck (4 ms)
[ RUN      ] BasicLinkedMapTests.RValueAccessCheck
[       OK ] BasicLinkedMapTests.RValueAccessCheck (5 ms)
[ RUN      ] BasicLinkedMapTests.LValueAccessCheck
[       OK ] BasicLinkedMapTests.LValueAccessCheck (3 ms)
[ RUN      ] BasicLinkedMapTests.ContainsCheck
[       OK ] BasicLinkedMapTests.ContainsCheck (4 ms)
[ RUN      ] BasicLinkedMapTests.EraseCheck
[       OK ] BasicLinkedMapTests.EraseCheck (6 ms)
[ RUN      ] BasicLinkedMapTests.KeyRangeCheck
[       OK ] BasicLinkedMapTests.KeyRangeCheck (6 ms)
[ RUN      ] BasicLinkedMapTests.SortedKeyCheck
[       OK ] BasicLinkedMapTests.SortedKeyCheck (5 ms)
[ RUN      ] BasicLinkedMapTests.InvalidKeyCheck
[       OK ] BasicLinkedMapTests.InvalidKeyCheck (5 ms)
[ RUN      ] BasicLinkedMapTests.NextKeyCheck
[       OK ] BasicLinkedMapTests.NextKeyCheck (7 ms)
[ RUN      ] BasicLinkedMapTests.PrevKeyCheck
[       OK ] BasicLinkedMapTests.PrevKeyCheck (7 ms)
[----------] 11 tests from BasicLinkedMapTests (59 ms total)

[----------] Global test environment tear-down
[==========] 22 tests from 2 test suites ran. (206 ms total)
[  PASSED  ] 22 tests.
==326661==
==326661== HEAP SUMMARY:
==326661==     in use at exit: 0 bytes in 0 blocks
==326661==   total heap usage: 702 allocs, 702 frees, 152,526 bytes allocated
==326661==
==326661== All heap blocks were freed -- no leaks are possible
==326661==
==326661== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
```

Performance test output:

```
# All times in milliseconds (msec)
# Column 1 = input data size
# Column 2 = array map insert
# Column 3 = linked map insert
# Column 4 = array map erase
# Column 5 = linked map erase
# Column 6 = array map contains
# Column 7 = linked map contains
# Column 8 = array map find range
# Column 9 = linked map find range
# Column 10 = array map next key
# Column 11 = linked map next key
# Column 12 = array map sorted keys
# Column 13 = linked map sorted keys
0 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
2000 0.00 0.00 0.03 11.06 0.02 9.48 0.02 5.75 0.01 5.06 0.22 5.42
4000 0.00 0.00 0.02 20.79 0.02 20.62 0.03 20.76 0.02 20.84 0.45 20.99
6000 0.00 0.00 0.03 47.11 0.03 46.82 0.06 48.74 0.04 46.91 0.72 47.21
8000 0.00 0.00 0.04 82.40 0.04 85.89 0.07 82.83 0.05 85.57 0.96 83.21
10000 0.00 0.00 0.05 129.17 0.05 129.66 0.09 134.07 0.07 129.40 1.19 129.56
12000 0.00 0.00 0.06 185.88 0.07 186.44 0.11 193.81 0.09 189.78 2.20 187.65
14000 0.00 0.00 0.08 255.28 0.07 254.85 0.12 255.70 0.10 257.80 2.58 254.22
16000 0.00 0.00 0.08 332.36 0.09 331.72 0.14 339.52 0.12 335.29 3.06 330.87
18000 0.00 0.00 0.10 419.87 0.10 422.70 0.15 438.13 0.14 421.27 3.69 421.42
20000 0.00 0.00 0.11 520.48 0.10 521.46 0.16 526.93 0.16 521.44 3.92 525.26
30.30user 0.01system 0:30.32elapsed 99%CPU (0avgtext+0avgdata 4696maxresident)k
0inputs+56outputs (0major+571minor)pagefaults 0swaps
```