

Name: *Samuel Sovi*

Assignment: 4

Complete & Correct:	31 / 42
Tests:	6 / 6
Format and Comments:	6 / 6
Writeup:	6 / 6

Total Score:	49 / 60
--------------	-------	---------

Comments:

1. (-3) your linked random quick sort calls quick sort helper (instead of the randomized version)
2. (-6) your array random quick sort has same issue
3. note that fixing the linked version speeds up your random quick sort significantly
4. (-2) changing your array random quick sort results in failed tests

Unit test output:

```

==692729== Memcheck, a memory error detector
==692729== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==692729== Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright info
==692729== Command: ./hw4_test
==692729==
[=====] Running 36 tests from 2 test suites.
[-----] Global test environment set-up.
[-----] 18 tests from BasicArraySeqTests
[ RUN      ] BasicArraySeqTests.EmptySeqMergeSort
[      OK  ] BasicArraySeqTests.EmptySeqMergeSort (11 ms)
[ RUN      ] BasicArraySeqTests.OneElemMergeSort
[      OK  ] BasicArraySeqTests.OneElemMergeSort (4 ms)
[ RUN      ] BasicArraySeqTests.TwoElemInOrderMergeSort
[      OK  ] BasicArraySeqTests.TwoElemInOrderMergeSort (5 ms)
[ RUN      ] BasicArraySeqTests.TwoElemReverseMergeSort
[      OK  ] BasicArraySeqTests.TwoElemReverseMergeSort (3 ms)
[ RUN      ] BasicArraySeqTests.ThreeElemMergeSortCases
[      OK  ] BasicArraySeqTests.ThreeElemMergeSortCases (3 ms)
[ RUN      ] BasicArraySeqTests.EmptySeqQuickSort
[      OK  ] BasicArraySeqTests.EmptySeqQuickSort (1 ms)
[ RUN      ] BasicArraySeqTests.OneElemQuickSort
[      OK  ] BasicArraySeqTests.OneElemQuickSort (2 ms)
[ RUN      ] BasicArraySeqTests.TwoElemInOrderQuickSort
[      OK  ] BasicArraySeqTests.TwoElemInOrderQuickSort (4 ms)
[ RUN      ] BasicArraySeqTests.TwoElemReverseQuickSort
[      OK  ] BasicArraySeqTests.TwoElemReverseQuickSort (3 ms)
[ RUN      ] BasicArraySeqTests.ThreeElemQuickSortCases
[      OK  ] BasicArraySeqTests.ThreeElemQuickSortCases (3 ms)
[ RUN      ] BasicArraySeqTests.EmptySeqRandQuickSort
[      OK  ] BasicArraySeqTests.EmptySeqRandQuickSort (3 ms)
[ RUN      ] BasicArraySeqTests.OneElemRandQuickSort
[      OK  ] BasicArraySeqTests.OneElemRandQuickSort (2 ms)
[ RUN      ] BasicArraySeqTests.TwoElemInOrderRandQuickSort

```

```

[      OK ] BasicArraySeqTests.TwoElemInOrderRandQuickSort (4 ms)
[ RUN      ] BasicArraySeqTests.TwoElemReverseRandQuickSort
[      OK ] BasicArraySeqTests.TwoElemReverseRandQuickSort (3 ms)
[ RUN      ] BasicArraySeqTests.ThreeElemRandQuickSortCases
[      OK ] BasicArraySeqTests.ThreeElemRandQuickSortCases (3 ms)
[ RUN      ] BasicArraySeqTests.FourElemQuickSortCases
[      OK ] BasicArraySeqTests.FourElemQuickSortCases (6 ms)
[ RUN      ] BasicArraySeqTests.FourElemRandQuickSortCases
[      OK ] BasicArraySeqTests.FourElemRandQuickSortCases (6 ms)
[ RUN      ] BasicArraySeqTests.FourElemMergeSortCases
[      OK ] BasicArraySeqTests.FourElemMergeSortCases (6 ms)
[-----] 18 tests from BasicArraySeqTests (90 ms total)

[-----] 18 tests from BasicLinkedSeqTests
[ RUN      ] BasicLinkedSeqTests.EmptySeqMergeSort
[      OK ] BasicLinkedSeqTests.EmptySeqMergeSort (2 ms)
[ RUN      ] BasicLinkedSeqTests.OneElemMergeSort
[      OK ] BasicLinkedSeqTests.OneElemMergeSort (4 ms)
[ RUN      ] BasicLinkedSeqTests.TwoElemInOrderMergeSort
[      OK ] BasicLinkedSeqTests.TwoElemInOrderMergeSort (5 ms)
[ RUN      ] BasicLinkedSeqTests.TwoElemReverseMergeSort
[      OK ] BasicLinkedSeqTests.TwoElemReverseMergeSort (3 ms)
[ RUN      ] BasicLinkedSeqTests.ThreeElemMergeSortCases
[      OK ] BasicLinkedSeqTests.ThreeElemMergeSortCases (4 ms)
[ RUN      ] BasicLinkedSeqTests.EmptySeqQuickSort
[      OK ] BasicLinkedSeqTests.EmptySeqQuickSort (1 ms)
[ RUN      ] BasicLinkedSeqTests.OneElemQuickSort
[      OK ] BasicLinkedSeqTests.OneElemQuickSort (2 ms)
[ RUN      ] BasicLinkedSeqTests.TwoElemInOrderQuickSort
[      OK ] BasicLinkedSeqTests.TwoElemInOrderQuickSort (4 ms)
[ RUN      ] BasicLinkedSeqTests.TwoElemReverseQuickSort
[      OK ] BasicLinkedSeqTests.TwoElemReverseQuickSort (3 ms)
[ RUN      ] BasicLinkedSeqTests.ThreeElemQuickSortCases
[      OK ] BasicLinkedSeqTests.ThreeElemQuickSortCases (4 ms)
[ RUN      ] BasicLinkedSeqTests.EmptySeqRandQuickSort
[      OK ] BasicLinkedSeqTests.EmptySeqRandQuickSort (1 ms)
[ RUN      ] BasicLinkedSeqTests.OneElemRandQuickSort
[      OK ] BasicLinkedSeqTests.OneElemRandQuickSort (2 ms)
[ RUN      ] BasicLinkedSeqTests.TwoElemInOrderRandQuickSort
[      OK ] BasicLinkedSeqTests.TwoElemInOrderRandQuickSort (4 ms)
[ RUN      ] BasicLinkedSeqTests.TwoElemReverseRandQuickSort
[      OK ] BasicLinkedSeqTests.TwoElemReverseRandQuickSort (3 ms)
[ RUN      ] BasicLinkedSeqTests.ThreeElemRandQuickSortCases
[      OK ] BasicLinkedSeqTests.ThreeElemRandQuickSortCases (4 ms)
[ RUN      ] BasicLinkedSeqTests.FourElemQuickSortCases
[      OK ] BasicLinkedSeqTests.FourElemQuickSortCases (6 ms)
[ RUN      ] BasicLinkedSeqTests.FourElemRandQuickSortCases
[      OK ] BasicLinkedSeqTests.FourElemRandQuickSortCases (6 ms)
[ RUN      ] BasicLinkedSeqTests.FourElemMergeSortCases
[      OK ] BasicLinkedSeqTests.FourElemMergeSortCases (6 ms)
[-----] 18 tests from BasicLinkedSeqTests (73 ms total)

[-----] Global test environment tear-down
[=====] 36 tests from 2 test suites ran. (204 ms total)
[ PASSED ] 36 tests.
==692729==
==692729== HEAP SUMMARY:
==692729==    in use at exit: 0 bytes in 0 blocks
==692729==   total heap usage: 1,117 allocs, 1,117 frees, 175,999 bytes allocated
==692729==
==692729== All heap blocks were freed -- no leaks are possible
==692729==

```

```

==692729== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)

==692731== Memcheck, a memory error detector
==692731== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==692731== Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright info
==692731== Command: ./hw4_test_extra
==692731==
[=====] Running 6 tests from 2 test suites.
[-----] Global test environment set-up.
[-----] 3 tests from ExtraArraySeqTests
[ RUN      ] ExtraArraySeqTests.MergeSortThenAddAndErase
[      OK  ] ExtraArraySeqTests.MergeSortThenAddAndErase (23 ms)
[ RUN      ] ExtraArraySeqTests.QuickSortThenAddAndErase
[      OK  ] ExtraArraySeqTests.QuickSortThenAddAndErase (8 ms)
[ RUN      ] ExtraArraySeqTests.QuickSortRandomThenAddAndErase
[      OK  ] ExtraArraySeqTests.QuickSortRandomThenAddAndErase (10 ms)
[-----] 3 tests from ExtraArraySeqTests (48 ms total)

[-----] 3 tests from ExtraLinkedSeqTests
[ RUN      ] ExtraLinkedSeqTests.MergeSortThenAddAndErase
[      OK  ] ExtraLinkedSeqTests.MergeSortThenAddAndErase (13 ms)
[ RUN      ] ExtraLinkedSeqTests.QuickSortThenAddAndErase
[      OK  ] ExtraLinkedSeqTests.QuickSortThenAddAndErase (8 ms)
[ RUN      ] ExtraLinkedSeqTests.QuickSortRandomThenAddAndErase
[      OK  ] ExtraLinkedSeqTests.QuickSortRandomThenAddAndErase (8 ms)
[-----] 3 tests from ExtraLinkedSeqTests (30 ms total)

[-----] Global test environment tear-down
[=====] 6 tests from 2 test suites ran. (119 ms total)
[ PASSED  ] 6 tests.
==692731==
==692731== HEAP SUMMARY:
==692731==       in use at exit: 0 bytes in 0 blocks
==692731==   total heap usage: 360 allocs, 360 frees, 127,251 bytes allocated
==692731==
==692731== All heap blocks were freed -- no leaks are possible
==692731==
==692731== 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 = avg time array merge sort, reversed
# Column 3 = avg time array merge sort, shuffled
# Column 4 = avg time array quick sort, reversed
# Column 5 = avg time array quick sort, shuffled
# Column 6 = avg time array quick sort random, reversed
# Column 7 = avg time array quick sort random, shuffled
# Column 8 = avg time linked merge sort, reversed
# Column 9 = avg time linked merge sort, shuffled
# Column 10 = avg time linked quick sort, reversed
# Column 11 = avg time linked quick sort, shuffled
# Column 12 = avg time linked quick sort random, reversed
# Column 13 = avg time linked quick sort random, shuffled
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
1500 0.00 0.00 5.00 0.00 2.00 0.00 0.00 0.00 6.00 0.00 4.00 0.00
3000 0.00 0.00 14.00 0.00 6.00 0.00 0.00 0.00 47.00 1.00 21.00 0.00
4500 0.00 0.00 33.00 0.00 24.00 0.00 0.00 0.00 105.00 2.00 98.00 1.00
6000 0.00 0.00 59.00 0.00 40.00 0.00 0.00 0.00 189.00 3.00 139.00 2.00
7500 0.00 0.00 92.00 0.00 67.00 0.00 0.00 0.00 309.00 5.00 252.00 5.00

```

9000	0.00	0.00	133.00	1.00	65.00	1.00	0.00	0.00	451.00	9.00	271.00	6.00
10500	0.00	1.00	180.00	1.00	110.00	1.00	0.00	0.00	641.00	13.00	488.00	11.00
12000	1.00	1.00	435.00	2.00	157.00	2.00	1.00	1.00	921.00	22.00	620.00	16.00
13500	2.00	2.00	552.00	3.00	298.00	2.00	1.00	1.00	1163.00	30.00	854.00	19.00
15000	2.00	2.00	691.00	3.00	581.00	3.00	1.00	1.00	1452.00	35.00	1555.00	26.00