

CS311 Yoshii - HW2 Part 2 – Sorting Analysis (based on week4 - week 5)

DUE: Week 6 Monday

TOTAL: 37 points **Your score is:**

Your NAME: Eduardo Martinez

Date Turned in: 10/3/16

Purpose: To demonstrate your understanding of analyzing searching and sorting algorithms.

A) Review Questions [1pt per question = 5pts] Your score is:
Type your answers here

1. Would you use Selection Sort or would you use Insertion Sort?
Insertion sort
 - Why?
Selection sort's $B(n)$ and $W(n)$ are both $O(N^2)$ while insertion sort $B(n)$ is $N-1$ and the $W(n)$ is the same and as selection sort. Which means that generally insertion sort does less comparisons than selection sort unless in the worst case scenario.
2. Rika-Chu Sort corrects one inversion per comparison.
 - What is its worst case number of comparisons?
The worst case scenario or $W(n)$ of Rika-Chu sort would be $(n-1)/2$
 - Why?
The worst case scenario would happen when the every pair in the list is a bad pair.
3. What is the advantage of using Merge Sort over Quick Sort?
The advantage of merge sort over quick sort is its speed.
4. What is the disadvantage of using Merge Sort over Quick Sort?
The disadvantage of using merge sort over quick sort is that it uses a lot of memory to sort .
5. Why is Radix sort unrelated to the $F(n) = O(n \log n)$ theorem?

Radix sort is unrelated to $F(n) = O(n \log n)$ theorem because it does not use comparisons.

B) Sort

230 123 324 10 23 56 (6 items)

using Insertion Sort and fill in the answers below.

[1/2 per prompt=7pts]

Your score:

Start with pos 2 for X index:

Which items were shifted?

123 and 230

How many element comparisons until X is deposited back?

1

The resulting list is?

123 230 321 10 23 56

Start with pos 3 for X index:

Which items were shifted?

None

How many comparisons until X is deposited back?

1

The resulting list is?

123 230 321 10 23 56

Start with pos 4 for X index:

Which items were shifted?

123 230 324 10

How many comparisons until X is deposited back?

3

The resulting list is?

10 123 230 324 23 56

Start with pos 5 for X index:

Which items were shifted?

23 123 230 324

How many comparisons until X is deposited back?

4

The resulting list is?

10 23 123 230 324 56

Start with pos 6 for X index:

Which items were shifted?

56 123 230 324

How many comparisons until X is deposited back?

4

The final resulting sorted list is?

10 23 56 123 230 324

Q) Total number of comparisons was (add up the above): 13

Q) Give an example list for which you would have made the worst number of comparisons:

324 230 123 56 23 10

C) Using the Merge Sort algorithm, sort [1/4 per prompt=9pts] Your score:
8 5 6 3 9 2 1 7.

Fill in the []'s:

1. Break this up into: [8 5 6 3] and [9 2 1 7]
2. Break these up into: [8 5] and [6 3] [9 2]
and [1 7]
3. Further Break these up into: [8] and [5] [6] and [3] [9] and [2] [1] and
[2]

For the one element lists:

Combine what and what?

8 5

Produce what?

5 8

How many element comparisons for this part?

1

Combine what and what?

6 3

Produce what?

3 6

How many comparisons for this part?

1

Combine what and what?

9 2

Produce what?

2 9

How many comparisons?

1

Combine what and what?

1 2

Produce what?

1 2

How many comparisons?

1

For the two element lists:

Combine what and what?

5 8 and 3 6

Produce what?

3 5 6 8

How many comparisons? 3

Combine what and what?

2 9 and 1 7

Produce what?

1 2 7 9

How many comparisons? 3

Final step:

Combine what and what?

3 5 6 8 and 1 2 7 9

Produce what? (the final result):

1 2 3 5 6 7 8 9

How many comparisons? 7

Q) Total number of comparisons was? (add up the above): 17

D) Sort

231 123 324 100 230 560 (6 items)

using Radix Sort.

Hint: use 0-list, 1-list, 3-list, 4-list etc. [1 per prompt=6pts] Your score:

Pass1:

Show the sub-lists here based on the last char

0: 100 230 560

1: 231

3: 123

4: 324

Show the combined list

100 230 560 231 123 324

Pass2:

Show the sub-lists here based on the second char

0: 100

2: 123 324

3: 230 231

6: 560

Show the combined list

100 123 324 230 231 560

Pass3:

Show the sub-lists here based on the first char

1: 100 123

2: 230 231

3: 324

5: 560

Show the combined list

100 123 230 231 324 560

E) Program Merge Sort's Combine: [2+8=10 pts] Your score:

Using Notes-4B.doc, code only the Procedure Combine of Merge Sort.

No ADT is needed. Just one source code file. – Run my solution program first.

Void Combine will take 3 vectors as arguments: A, B and R.

Combine should work for any size vectors as long as the size of A and B are the same.

It will combine the elements of A and B into R to produce the sorted list R.

You should know how to find the size of a vector.

Display "comparison" every time an element-element comparison is done.

Your main()

- 1) Will declare three vectors L1, L2 and L3.
- 2) Will ask the user to type integers in increasing order into L1.
- 3) Then ask the user to type more integers in increasing order into L2.
- 4) Then it will call void Combine to combine L1 and L2 to produce L3 which is passed back by reference.
- 5) Display what is in L3.

Required Test Cases: (Must test in this order)

- 1) Combine 1 2 3 with 4 5 6
- 2) Combine 1 3 5 with 2 4 6
- 3) Combine 4 5 6 with 1 2 3
- 4) Combine 1 2 5 6 with 3 4 7 8

Q) State of the program [2pts]

- Does your program compile without errors? If not, describe: yes
- List any bugs you are aware of, or state "No bugs": no bugs

Submit these 3 files:

- 1) This assignment sheet with inserted answers.
- 2) Source code file (**combine.cpp**) of the program (with good comments).
- 3) Script (**Test**) of the compilation and test results.

Did you answer all the questions?

