roblem 1

Show that any comparison-based algorithm to sort 4 elements requires at least 5 comparisons in the worst case. Decision tree has the least comparisons and the rest comparison-based algorithms have at least comparisons as decision tree. For four elements, there are 4! permutations, and for the worst case, there are at least log(4!) comparisons needed which is 5.

Problem 2
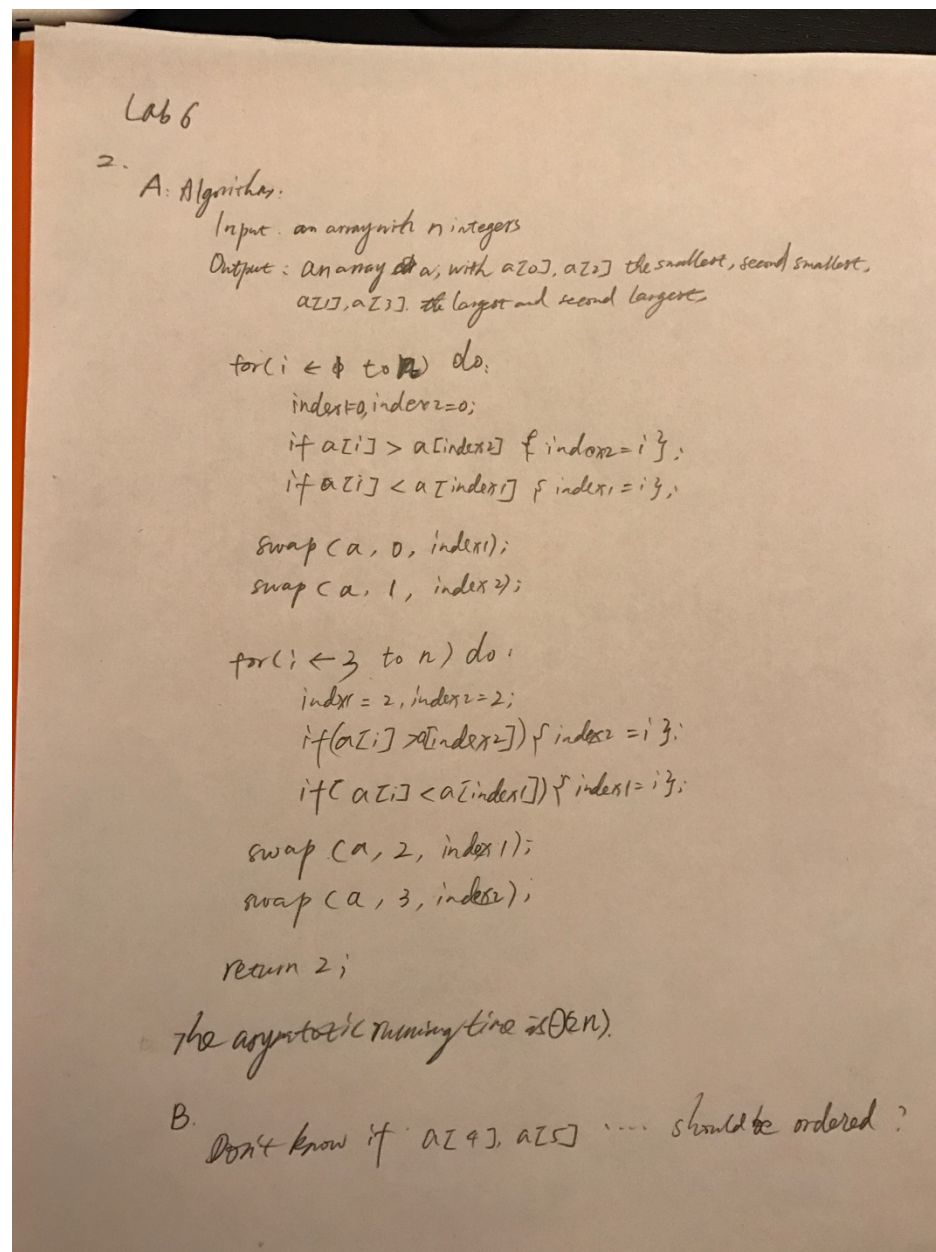
Problem 3

3. Radix Sort $\{80, 27, 72, 1, 27, 8, 64, 34, 16\}$. Use 9 for radix

$K = 9q + r$ where $0 \leq r < 9$, $0 \leq q < 9$.

$80 = 9 \times 8 + 8$
$27 = 9 \times 3 + 0$
$72 = 9 \times 8 + 0$
$1 = 9 \times 0 + 1$
$27 = 9 \times 3 + 0$
$8 = 9 \times 0 + 8$
$64 = 9 \times 7 + 1$
$34 = 9 \times 3 + 7$
$16 = 9 \times 1 + 7$

pass #1. bucket 1:   remainders bucket.

pass #2. Bucket 2:   quotients bucket

Output: 1, 8, 16, 27, 27, 34, 64, 72, 80

Asymtotic running time is: $\Theta(n)$

PS: any two number with same quotation, the one with smaller remainder is processed in front of another one with larger remainder, this is the reason why we put the first one in front of the second in the first bucket.

The first bucket store the numbers using the remainder as the index. the second bucket stores the numbers using the quotients as the index.

quotient $\frac{2}{3}$ quotient

Problem 4

4. 1. create a bucket with size $3n$, $b[3n]$.

2. Iterate over the input array, a

$$\text{set } b[a[i]] = \begin{cases} 1 & (b[a[i]] = 0) \\ i+1 & (b[a[i]] = 1) \end{cases}$$ $O(n)$

3. Iterate over the bucket,

if $b[i] = 1$, return i $O(3n)$

$\Rightarrow T = O(n) + O(3n) = O(n)$