Module 5 Problem Set Hsuan-You(Shaun) Lin

1. My Answer: O(n log(n))
My reason:
If we divide input into"M" parts each time in merge sort. complexity will be T(n) = Mn(log(n) to the base M), M is equal to 3 for this question, and usually M = 2 or 3 will be an efficient merge sort.
So according Master's Theorem:
=> $T(n) = 3T(n/3) + O(n) = O(n \log_3(n))$

The Complexity is $\underline{O(n \log(n))}$ #

---

2. My Answer: O(n log(n))
My reason:

Top Level: Even Divide

$$T_e(n) = 2T_e\left(\frac{n}{2}\right) + O(n)$$

Alt Level: Bad Divide

$$T_b(n) = T_b(n-1) + T(1) + O(n)$$
$$= T_b(n-1) + O(n)$$

=> Put these together:

$$\rightarrow T_b\left(\frac{n}{2}\right) = T_e\left(\frac{n}{2}-1\right) + O\left(\frac{n}{2}\right)$$
$$\rightarrow T_e(n) = 2T_e\left(\frac{n}{2}-1\right) + O\left(\frac{n}{2}\right) + O(n)$$
$$\rightarrow T_e(n) = 2T_e\left(\frac{n}{2}-1\right) + O(n)$$
$$\because T_e\left(\frac{n}{2}-1\right) \leq T_e\left(\frac{n}{2}\right)$$
$$\therefore T_e(n) \leq 2T_e\left(\frac{n}{2}\right) + O(n)$$

Therefore, by master's theorem => $T_e(n) = \underline{O(n \log(n))}$ #

---

3. My Answer: Insertion sort
My reason:

The worst case of Quicksort is $O(n^2)$, when sorted, or reverse sorted.

The expected complexity of Quicksort is $O(n \log(n))$

The worst case of insertion sort is $O(n^2)$

Alternative complexity mesure for insertion sort is O(# inversions), so could get O(n) when #inversion is O(n)

∴ I will choose <u>insertion sort</u> # to sort the list by date.

---

4. My Answer: $O(k^2 n)$
My reason:

n n n n ... n
↑—— k ——↑

Step 1 : merge (1, 2) Work = 2n
Step 2 : merge (1', 2') Work = 2n + n = 3n
⋮
Step k−1 : merge (1'''..., 2'''...) Work = (k−1)n + n = kn

Total: $2n + 3n + 4n + \ldots + kn = (2+3+4+\ldots+k)n = O(k^2 n)$

∴ The complexity of this n-way merge is $\underline{O(k^2 n)}$ #

5. My Answer: $O(kn \log(k))$
My reason

Phase 1: Merge pairs

Phase 2: Merge $\frac{k}{2}$ 2n

Work phase 1: Merge pairs $\frac{k}{2} 2n = kn$

Work phase 2: Merge $\frac{k}{4}$ pairs cost $4n = kn$

$\vdots$

work in any phase: kn

phase $= \log_2(k)$

Therefore, total work $= O(kn \log(k))$

$\therefore$ The complexity of this merge operation is $\underline{O(kn \log(k))}$ #