(5) Computing the time complexity.

↳ Assign cost and no. of times of execution of each stmt.

Insertion sort A[1..n].	cost	times
1. for $j \leftarrow 2$ to length[A]	c_1	n
2. do key $\leftarrow A[j]$	c_2	$n-1$
3. // Insert A[j] into sorted A[1..j-1]	0	$n-1$
4. $i \leftarrow j-1$	c_4	$n-1$
5. while $(i>0)$ and $A[i] >$ key	c_5	$\sum_{j=2}^{n} t_j$
6. do $A[i+1] \leftarrow A[i]$	c_6	$\sum_{j=2}^{n} t_j - 1$
7. $i \leftarrow i-1$	c_7	$\sum_{j=2}^{n} t_j - 1$
8. $A[i+1] \leftarrow$ key.	c_8	$n-1$

$$T(n) = c_1 n + c_2(n-1) + c_4(n-1) + c_5 \sum_{j=2}^{n} t_j + c_6 \sum_{j=2}^{n} t_j - 1 + c_7 \sum_{j=2}^{n} t_j - 1 + c_8(n-1)$$

Best case analysis:

↳ For each value of j (ie) $j = 2, 3 \cdots n$, $A[i] > key$ cond fails. when i has its initial value of $j-1$. ~~(ie) in the first iterati~~

↳ Thus $t_j = 1$ for each value of j

$$\therefore \sum_{j=2}^{n} t_j = \sum_{j=2}^{n} 1 = n-1$$

$\therefore$ $(n-1)$ times while loop stmt is executed.

↳ line 6 & 7 are not at all executed.

$$T(n) = c_1 n + c_2(n-1) + c_4(n-1) + c_5(n-1) + c_8(n-1)$$
$$= (c_1 + c_2 + c_4 + c_5 + c_8)n - (c_2 + c_4 + c_5 + c_8)$$
$$= an + b \text{ where } a \text{ \& } b \text{ are consts.}$$

$\therefore$ Best case complexity is linear fn of n

$\therefore O(n)$.

Worst case analysis:

↳ For each value of j, the element A[j] is compared with each and every element in the sorted subarray A[1 ⋯ j−1].

↳ So $t_j = j$ for each value of j

$$\sum_{j=2}^{n} t_j = \sum_{j=2}^{n} j = \frac{n(n+1)}{2} - 1$$

$t_j = j$ then $t_j - 1 = j - 1$

↳ If ~~while loops is executed for~~

$$\sum_{j=2}^{n} t_j - 1 = \sum_{j=2}^{n} j - 1 = \frac{n(n-1)}{2}.$$

$$T(n) = c_1 n + c_2(n-1) + c_4(n-1) + c_5\left(\frac{n(n+1)}{2} - 1\right)$$

$$+ c_6\left(\frac{n(n-1)}{2}\right) + c_7\left(\frac{n(n-1)}{2}\right) + c_8(n-1)$$

$$= \left(\frac{c_5}{2} + \frac{c_6}{2} + \frac{c_7}{2}\right)n^2 + \left(c_1 + c_2 + c_4 + \frac{c_5}{2} - \frac{c_6}{2} - \frac{c_7}{2} + c_8\right)n$$

$$- (c_2 + c_4 + c_5 + c_8)$$

$$= an^2 + bn + c \quad \therefore a, b, c \text{ are consts.}$$

Week 7

$\therefore$ worst case complexity is quadratic function of n.

$\therefore O(n^2)$.