CSE 5050: Complexity Analysis Handout

Example of complexity analysis

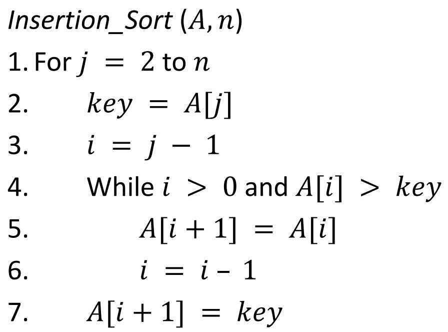
In the video lectures of Module 2, we saw how to analyze the worst-case time complexity of Gale-Shapley. Here, we show how to analyze the worst-case time complexity of Insertion Sort.

Question

Provide a tight bound on the worst-case time complexity of Insertion Sort.

Solution

Consider the pseudocode for insertion sort:



```
Insertion_Sort ($A, n$)
1. For $j = 2$ to $n$
2.     $key = A[j]$
3.     $i = j - 1$
4.     While $i > 0$ and $A[i] > key$
5.         $A[i + 1] = A[i]$
6.         $i = i - 1$
7.     $A[i + 1] = key$
```

We will first compute an upper bound on the worst-case time complexity of this algorithm. Observe that each of the 7 steps in the algorithm requires only $O(1)$ time per execution. We therefore focus on determining the number of times each step is executed. Let t_j denote the number of times the 'While' loop is executed for the j-th element in the input array. Then,
Step 1 is executed $O(n)$ times,
Step 2 is executed $O(n)$ times,
Step 3 is executed $O(n)$ times,
Step 4 is executed $\sum_{j=2}^{n} t_j$ times,
Step 5 is executed $\sum_{j=2}^{n}(t_j - 1)$ times,
Step 6 is executed $\sum_{j=2}^{n}(t_j - 1)$ times, and
Step 7 is executed $O(n)$ times.

Now, since t_j cannot exceed j, we must have $\sum_{j=2}^{n} t_j \leq 2 + 3 + \ldots + n$, which is $O(n^2)$. Thus, Steps 4, 5, and 6 are each executed $O(n^2)$ times. Summing up these complexities over all seven steps, we get a total time complexity of $O(n^2)$ for Insertion Sort.

We will now show that the worst-case time complexity of Insertion sort must be $\Omega(n^2)$. Consider what happens when the input is in reverse sorted order. In this case, we get $t_j = j$, since the j-th number must be moved all the way to the beginning of the list. Thus, in this case, $\sum_{j=2}^{n} t_j = 2 + 3 + \ldots + n$, which is $\Omega(n^2)$.

Since the worst-case time complexity of Insertion sort is both $O(n^2)$ and $\Omega(n^2)$, it must be $\Theta(n^2)$. This gives a tight bound on the worst-case time complexity of Insertion Sort, as required.