C-8.1

In order to the find the kth smallest key in the union of the keys from S and T, we can do a "double" binary search. In other words, we will begin by examining the $k/2$th element in the sequence S. Next, we will find the largest element in T that is less than $S[k/2]$ by binary search. Then, we will add the indices of the elements we were examining in S and T. If the sum equals k, then the max of the two elements is our result. If the sum is greater than k, then we will do a binary search to the right (upwards) in S. If the sum is less than k, then we will do a binary search to the left (downwards) in S. This is followed once again by searching in T for largest element less than the current element in S, etc. This method does a binary search on S which requires $O(\log n)$ "probes." However, for each probe of the search, it does a binary search on T which takes $O(\log n)$ time. Thus, the entire method takes $O(\log^2 n)$ time.