Problem 4.1 (merge sort)
d) (Bonus)

- In the majority of the cases, k (average and worst)
should be ≤ 20 as it is the one with
the least computation time as stated in C (answer)
best case (sorted already or nearly sorted)
k is best taken as the size of the
array or the number of elements of
the array as n is less than $n \lg n$ and for $n \geq 2$
it makes for less computation time
as it directly goes to insertion sort because
$n = k$ where $O(n)$ is best case for
insertion sort and $O(n \lg n)$ is best case
for merge sort. (computation times)
This happens because this merge sort
variant goes to insertion sort when it
reaches a certain value k and if $n = k$, it
goes straight to insertion sort which
has $O(n) \leq O(n \lg n)$ best case time complexi-
(insertion) (merge sort)
-ty which makes for less computation
time.