Ex 2.1.19: Inversions. Develop and implement a linearithmic algorithm for computing the number of inversions in a given array (the number of exchanges that would be performed by insertion sort for that array—see Section 2.1). This quantity is related to the Kendall tau distance;

Solution: Since we want a linearithmic solution for counting number of inversions in an array, we can use mergesort. As we sort the array by recursively breaking it down, we can count the number of inversions at each step. Then when we start to merge the arrays back together, we can count the number of inversions in the combined array.

So to summarize, as we sort the array, we can count the number of inversions at each recursively broken down step. This leads to a time complexity of $O(nLogn)$.