CS5008 Homework 7 Merge Sort

Written Exercises

Name: Minjie Shen

1. Explain what you think the worst-case, big-Oh complexity and the best-case, big-Oh complexity of merge sort is. Why do you think that?

The worst case and best case big-Oh complexity of merge sort is O(N*LogN). From the understanding of the merge sort implementation, we have to repeatedly split the array into half part, and for a size of N array, we need to split LogN times (at most log N + 1 times). And for the merge step, we need to iterate over every element to decide the sequence. In total, the time complexity is O(N * logN).

2. Explain what you think the worst-case, big-Oh complexity and the best-case, big-Oh complexity is for this iterative merge sort. Why do you think that?

For the iterative approach of merge sort, the time complexity in big-Oh is O(N* logN) . The first for loop runs for logN times since the m increase by 2 multiply by m each time. The inner for loop and the merge method total takes N times since it must compare every position in the original array and total is N times.