1. Given an array of size n, what do you think the worst case running time of treeSort() is? How did you get this answer? What input could yield this running time?
   1. nlog(n) I think. I would get this because I am calling my equation recursively and not iteratively. Recursion creates the log(n) in this equation. Input that would cause this running time would be anything. Random input is good.
2. Given an array of size n, what do you think the best case running time of treeSort() is? How did you get this answer? What input would be needed to yield this running time?
   1. nlog(n). I got this answer through researching online. Being that it recurses though, the quickest it can become is nlog(n). Its biggest disadvantage is also its advantage. Random input is good.
3. Can you think of any circumstances where treeSort() may make more sense than mergeSort() or quickSort()?
   1. When space is limited, a treeSort() might be a good idea.