# CSC236 Name: Tradd Schmidt

List the results you obtained for the three runs of linear search and sort + binary search with the time.time() function using war\_and\_peace.txt. Reflect on these results and whether or not they are what you expected.

|  |
| --- |
| War: linear took 0.015630006790161133 seconds to complete and the binary plus sort took 0.3281714916229248 seconds  Aah: linear took 0.015598773956298828 seconds to complete and the binary plus sort took 0.3906741142272949 seconds  Shawl: linear took 0.015631675720214844 seconds to complete and the binary plus sort took 0.31253671646118164 seconds  I actually expected the binary search to be faster, but it turns out that the linear search is faster. |

Binary search requires that the list of words be sorted before it can work, and the sorting algorithm itself is not fast if the list is large. Its execution time can typically vary between ϴ(n2) and ϴ(n log(n)), which is often higher than the performance of linear search, which is ϴ(n).

When would the fact that the list of words must be sorted before using binary search be a disadvantage?

|  |
| --- |
| When the list is large and it takes the sort mechanism longer to complete than the linear search. |

In what situations can you reduce the time cost of sorting to the performance of binary search?

|  |
| --- |
| When you can have the words sorted before running the program. |

Suppose you decided to use both the linear and binary searches onto a sorted list to get them on an even playing field, so to speak. When would linear search always outperform binary search? When would binary search performance be faster? Explain.

|  |
| --- |
| If the list is ordered sequentially, when the target word is extremely close to the beginning of the set of words linear will be faster. Any other time binary search will be faster. The binary search will have to make a few guesses to find out that the target word is towards the beginning of the set. In the meantime, the linear search will already have found the target word. |