Search Test Lab Report

Names:

Zhuocai Li

**1. Linear Search**

We know from class that the theoretical time complexity of linear search over *unordered lists* is:

|  |  |  |
| --- | --- | --- |
| **Best Case** | **Worst Case** | **Average Case** |
| *1* | *N* | *N/2* |

**Q1:** Increasing the number of trials and the value of N

1. Run experiments with an increasing value of N (from 1000 to 10,000). Does increasing N affect how many trials you have to run to get accurate results? Explain.

**The number of trial only helps to increase the accuracy of the final results, and can lead our results much closer to the ideal stage. If the value of N has increased, the trial should also increase in order to ensure the accuracy of the final output.**

1. Write down the number of trials that seem to have worked well for N=10,000.

|  |
| --- |
| **Number of Trials** |
| 5000 |

**Q2:** Linear Search Time Complexity Plot (Unordered List)

|  |
| --- |
|  |

**Q3:** Does the order of the data in the list affect the number of comparisons? In the table below, guess the time complexity of Linear Search on an *Ordered List.*

|  |  |  |
| --- | --- | --- |
| **Best Case** | **Worst Case** | **Average Case** |
| *1* | *N* | *N/2* |

Linear Search Time Complexity Plot (Ordered List)

|  |
| --- |
|  |

**Conclusion:**

**No matter in an ordered list or an unordered list, the time complexity for linear search is the same, with 1 for the best case, N for the worse case, and N/2 for the average case.**

**2. Binary Search**

We know from class that the theoretical time complexity of binary search over *ordered lists* are:

|  |  |  |
| --- | --- | --- |
| **Best Case** | **Worst Case** | **Average Case** |
| *1* | *log\_2(N)* | *log\_2(N)* |

**Q4:** Binary Search Time Complexity Plot

|  |
| --- |
|  |

**Conclusion:** What do your results tell you about the average-case complexity of Binary Search?

**The average-case complexity is almost the same as the worst case, which is log\_2(N).**

**3. Median**

Q5: We hypothesize that the time complexity of find\_median is:

|  |  |  |
| --- | --- | --- |
| **Best Case** | **Worst Case** | **Average Case** |
| *N* | *N^2* | *N^2/2* |

**Justification:**

1. Best case scenario:

*Happens when...*

**In the first loop, when i == 1, and it finds out the median number, then j will be equal to N, the total comparisons will be 1\*N = N.**

1. Best case scenario:

*Happens when...*

**In the last loop, when i == N, and it finds out the median number. For each time, j will be equal to N, the total comparisons will be N\*N = N^2.**

1. Average case scenario:

**The average case will be that the total amount of comparisons divides the total amount of trials. (1+2+3+…+N) \* N / N = (1+N) \* N / 2 This is almost the same as the worst case.**

Find\_median Time Complexity Plot

|  |
| --- |
|  |

**Conclusion:** Did your results support your hypothesis? If not, why not, and how does it change your original hypothesis?

**Yes. The best case will be N, the worse case will be N^2, while the average case will be N^2/2, according to the result of the experiments.**