**Binary Search Algorithm**

Barış Kaplan

Department of Computer Engineering, Koç University

ACWR 106: Academic Writing For Science & Technology

Sarah Elizabeth Kılınç

June 26, 2020

The binary search algorithm, one of the most used computer algorithms such as linear search and bubble sort, is a searching algorithm finding the user-given target value’s position in a sorted integer array, a fixed-sized and ordered list of integers, by using the divide and conquer approach. This approach involves recursively breaking down the problem into the more subproblems until the problem becomes easy to deal with. In 1946, John Mauchly started to use the binary search term. After some technological improvements, Derrick Henry Lehmer released a binary search algorithm working for all arrays in 1960.

There are several steps to apply this algorithm. First, we should take a search value from the user. Then, we should find the middle value in the array and compare the search value with the middle value. If they are equal, we should stop the algorithm. Otherwise, we should check if the search value is greater or smaller than the middle value. Subsequently, if the search value is smaller than the middle value, we should repeat the steps explained in the left sub-array. Otherwise, we should repeat the steps explained in the right sub-array. We should repeat the same process until the target value matches with the found value. If the target value is not equal to the last found value, we should give the message “Value is not found”, and terminate the algorithm.

Searching algorithms are basically divided into two categories, binary search and linear search. These algorithms have a main purpose of solving some problems. For example, we can find the median, the middle term of a sorted integer array, by using the binary search algorithm. There is a term called time complexity, used to estimate the run times of algorithms in the worst case. The worst case corresponds to the situation of maximum comparison. To find the target value, we do one-by-one comparisons for the linear search; however, we do binary comparisons for the binary search. So, for the sorted arrays, if we consider the worst case, the comparison number and the time complexity of the binary search are less, compared to the linear search. By taking this information into consideration, we conclude that in the worst case, finding the target values in the sorted arrays is faster and easier by using the binary search.

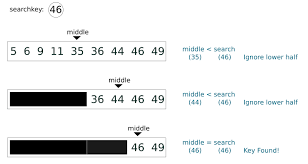


FIGURE 1: An example of the application of the binary search algorithm

References

(n.d.). How to implement binary search in python? Retrieved June 26, 2020, from

https://web.itu.edu.tr/temiz/