

HASH BASED SEARCHING ALGORITHM

Jyotirmayee Rautaray¹, Raghvendra Kumar²

School of Computer Engineering, KIIT University, Odisha, India¹

School of Computer Engineering, KIIT University, Odisha, India²

Abstract: Over the years, computer scientist participated research on linear search and Binary Search to find that which one is faster. These are two searching technique the condition of binary search array must be sorted order but in linear search that is not necessary. But in this paper proposed a new searching algorithm which is applicable for both sorted and unsorted array because in this paper we proposed Hash based searching technique for searching elements in the given array.

Keywords: Searching Technique, Linear Search, Binary Search, Hash Based Searching technique.

I. INTRODUCTION

Searching particular elements is very common task in computer algorithm. But in the real life application the number of array elements represents an object in the array. The main goal of any searching algorithm is find the particular searching elements that the user wants to search. But the main two distinct task of any searching algorithm one is process to storing of key and another is the process of retrieval of key, sequential searching algorithm is a very simple algorithm in which if the searching element that want to search is found then it's come and give that searching elements if not then that search till last elements of the array. This algorithm is applicable when the number of elements is very less in the array. And in the Binary search the condition is that array must be in sorted order. Binary search algorithms are based on the divided and conquer method so that searching time is very less as compare to other searching techniques. And it's also applicable for large number of elements.

II. LINEAR SEARCH ALGORITHM

The simplest method of searching of an element in a linear search [1] [2] [3] [4] [7]. Which simply check the first item and then second like on search all the item present in the array? Algorithm of linear search is as follow

Algorithm1
Step1:- Location = -1
Step2:- where I=0
Step3:- while (List[i] != target) and (I ≤ List length)
 I=I+1
Step4:- If List (I==Target)
Step5:- Location=I

For small number of array elements linear search is very faster as compare to the other searching algorithm. There are mainly two type of linear search one is order linear search and another is unordered linear search. The advantages of linear search are: linear search examine each list item until the target is found or the last elements of the array. The complexity of linear search in worst case is (N) and in best case is (1).

III. BINARY SEARCH ALGORITHM

The strategy of Binary search [5] [6] [7] [8] [9] is to check the middle of the elements in the list. If the key value if less than the middle value then it search to the left side till 0. And this process continues till N if the key value if greater than the middle value. The complexity of Binary is Log (N).

Here is main step of Binary Search algorithm is as follow

Algorithm2

/* given an array A (1, N) elements in the array in non decreasing order*/

/*N \geq 0, determining if X is present, such that X=A (J)*/

/*else J=0*/

Step1:-Integer Low, High, Mid, J, N

Step2:-Low \leftarrow 1, High \leftarrow N

Step3:-While Low \leq High Do

Step4:-Mid \leftarrow [(High + Low)/2]

Step5:-X < A (Mid), High \leftarrow Mid-1

Step6:-X > A (Mid), Low \leftarrow Mid+1

Step7:-Else J \leftarrow Mid

Step8:-Return

Step9:- Repeat Until J \leftarrow 0

IV. HASH BASED SEARCHING ALGORITHM

Hash Based Searching Algorithm[6] [7] indicate each array elements with a particular index value and hash that index value with the function in the form of $M \text{ mod } N$ (M is the index value and N is the user defined number). Hash based searching Algorithm is not depends upon the order or unordered list. And then map the hash value with the index value and then define the particular address to the elements. And then define the address to all value to the particular value. So that this increases the searching speed of the algorithm the complexity of the hash base searching technique is less than $\log(N)$. The algorithm of Hash based searching Algorithm is as follow shown in Algorithm3.

Algorithm3

/*Give an Array A (0, N) of number of elements*/

Step1:- Index Array from 0 to N-1 or 1 to N /* Defined by user*/

Step2:- Define a Hash function in the form of $M \text{ mod } N$ /* M- index number and N- User defined value*/

Step3:- Calculate a Hash value from 0 to N-1.

Step4:- Map the Hash value with the Index value. /*Mapping is done through row & column manner*/

Step5:-Address each array elements to the Index and Hash value.

Step6:- Give a particular address to each elements of an Array.

Step7:- Search an element with the particular address.

Step8:- End of Algorithm.

TABLE1- Elements of the array and Hash value

INDEX ID	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
ARRAY	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
HASH VALUE	0M od5 =0	1M od5 =1	2M od5 =2	3M od5 =3	4 Mo d5 =4	5M od5 =0	6M od5 =1	7M od5 =2	8M od5 =3	9M od5 =4	10 Mo d5 =0	11M od5 =1	12 Mo d5 =2	13M od5 =3	14M od5 =4	15M od5 =0	16M od5 =1	17M od5 =2	18M od5 =3	19M od5 =4

TABLE2- Index and Hash value

Index/ Hash Value	0	1	2	3
0	1	6	11	16
1	2	7	12	17
2	3	8	13	18
3	4	9	14	19
4	5	10	15	20

So that address 1 is 01, 2 is 10, 3 is 20.....20 is 43.

V. CONCLUSION

In this paper we proposed a new searching Algorithm and compare the complexity of existing one and proposed one. The complexity of linear search is N and binary search is $\log(N)$ because it used divide and conquer method but in the proposed one that used hash function so the complexity of the searching is very less as compare to the other existing one. So in future we implement an algorithm whose complexity is less the both these algorithm for order and unordered list.

REFERENCES

- [1]. Booch G., *Object Oriented Analysis and Design*, second Edition, Addison-Wesley, 1995.
- [2]. Collins W. J., *Data Structures*, first Edition Addison-Wesley publishing company, page 397,1992, U.S.A.
- [3]. Donald K., *The Act Computer Programming*, Volume 3, Sorting and Searching, Third Edition, Addison-Wesley, 1997.
- [4]. Horowitz E. and Sahni, *Fundamental of Computer Algorithms*, Computer science press, Inc, Maryland, 1978, U.S.A.
- [5]. Josuttis N. M., *The C++ Standard Library; A Tutorial and Reference*, Addison-Wesley, Reading, M.A, 1999.
- [6]. Kruse R. L., *Data Structures and Program design in C++*, Prentice Hall, page 280, 1999, USA.
- [7]. Rodger J., *Searching Arrays: Algorithms and Efficiency linear versus Binary Search*, Rai University, 2007.
- [8]. <http://en.wikipedia.org/wiki/binarysearchalgorithm>, last accessed on September 20, 2008.
- [9]. Stroustrup B., *The Design and Evolution of C++*, Addison-wesley Reading M. A, 1994.

BIOGRAPHY



Jyotirmayee Rautaray
 School of Computer Engineering, KIIT University, Bhubaneswar, Odisha, India



Raghvendra Kumar
 School of Computer Engineering, KIIT University, Bhubaneswar, Odisha, India