PAPER • OPEN ACCESS

Comparison Searching Process of Linear, Binary and Interpolation Algorithm

To cite this article: Robbi Rahim *et al* 2017 *J. Phys.: Conf. Ser.* **930** 012007

View the article online for updates and enhancements.

Related content

- A simple conversion formula for type 'T' (copper-constantan) thermocouple readings
  M R Boudry

- A new fractal-interpolation algorithm for seismic data based on iterated function systems
  Ming-Yue Zhai, Heidi Kuzma and James W. Rector

- Developing web-based data analysis tools for precision farming using R and Shiny
  Ebrahim Jahanshiri and Abdul Rashid Mohd Shariff

Recent citations

- Textbook Design on Temperature and Thermodynamic Based on Problem Solving Blended Learning Model
  Agus Rohman and Trise Nurul Ain

- Data Mining of Family, School, and Society Environments Influences to Student Performance
  Yo Ceng Giap *et al*

- Bipolar function in backpropagation algorithm in predicting Indonesia's coal exports by major destination countries
  Bayu Febriadi *et al*

Comparison Searching Process of Linear, Binary and Interpolation Algorithm

Robbi Rahim*[1], Saiful Nurarif[2], Mukhlis Ramadhan[2], Siti Aisyah[3] and Windania Purba[3]

[1]Department of Health Information, Akademi Perekam Medik dan Infokes Imelda, Jl. Bilal Ujung Medan 20116, Indonesia
[1]Student Universiti Malaysia Perlis, Universiti Malaysia Perlis, Malaysia
[2]Department of Computer System, STMIK Triguna Dharma, Jl. Jenderal Abdul Haris Nasution No.73, Medan 20219, Indonesia
[3]Department of Information System, Prima Indonesia University, Jl. Sekip Simpang Sikambing, Medan 20111, Indonesia

*usurobbi85@zoho.com

Abstract. Searching is a process that cannot be issued for a transaction and communication process, many search algorithms that can be used to facilitate the search, linear, binary, and interpolation algorithms are some searching algorithms that can be utilized, the comparison of the three algorithms is performed by testing to search data with different length with pseudo process approach, and the result achieved that the interpolation algorithm is slightly faster than the other two algorithms.

1. Introduction

Searching is a process that is usually performed in everyday activities, the search process is usually done to look for something with the purpose only to know whether the data is in a set of data or not [1] [2] [3], while at other times we may want the position of the data sought [3] [4] [5].

In computer science, there are various algorithms for search methods, such as the linear search method (Linear/Sequential Search), binary search and interpolation search [6]. Each algorithm has different prerequisites and different ways and execution time [6] [7]. The choice of search method meant on the user's circumstances, and the desire of the method which usually depends on the amount of data, data type and data structure used [8] [9].

This research tried to compare linear, binary and interpolation search algorithms with pseudo process approach for a gradual testing process.

2. Theory

Searching is a process with the aim to know whether the data is in a set of data or not, or maybe at other times the position of the data sought is needed for a particular purpose [1] [2] [3], or if the appearance of data more than once then all positions and frequency of appearance to be displayed [1] [2] [10], The simplest search can be described as follows [1]:

Suppose a sequence of data X [1] ... X [n], then the problem is whether Y (any data with the same data type as the existing data type in the X sequence and usually in input or already known) X [1] ... X [n] or whether Y is inside or not with Boolean result true / false or success / fail.

In computer science there are many algorithms for search method; generally, the data search methods split into several sections [4] such as:

1. Linear search method.
2. Binary search method.
3. Interpolation search method.

Each search algorithm has different prerequisites and ways and time of implementation, the selection of search methods is based on the circumstances, and desires of the user of the method which usually depends on the amount of data, data type and data structure used [8].

A. Linear Search Method

Linear Search performed on sequences of numbers that are ascending or descending or unordered. The linear search was given by comparing the data sought (X) with data in row A [1] ... A [n] starting with the first element data in sequence A. If the comparison is equal, then the search is stopped and declared successful [4] [6] [7]. Whereas if the comparison is not the same value then,

1. If the data not sorted (random data), then the search will proceed to the next data.
2. If the data sequentially sorted, then the search process will proceed to the next data which is to the right of the data, given that the data sought (X) is greater than the data being compared.

If data is descending, then the search will only continue to the next data which is to the right of the data being compared if the data sought (X) is smaller than the data being compared.

B. Binary Search Method

Binary Search can only perform on sequences of numbers that have been sorted either ascending or descending. Binary Search performs a search of X data in row A [1] ... A [n] with starting from intermediate data in sequence A. If X data value is equal to the middle data value of row A [4] [5], then the search is stopped and declared success. While if not the same then,

1. For ascending data, the search will be continued to the left ½ if the data value of X is smaller than the middle data value in sequence A. whereas if the data value of X is greater than the middle data value in sequence A, Will continue to ½ right.
2. For descending data, the search will be continued to the left ½ if the X data value is greater than the middle data value in sequence A. whereas if the X data value is less than the middle data value in sequence A, Will continue ½ left.

The search will be stopped and declared to fail if ½ left or right half is a single data and the data is not the same as the X data being searched [5].

C. Interpolation Search Method

Interpolation search looks for data by guessing (guessing) the position of the data searched by using a certain formula. Interpolation search can only perform on rows of numbers that have been sorted either ascending or descending [9] [10]. The alleged position as the data position sought (X) can be calculated using the following formula:

$$Pos = ceiling [BB + ((X - A [BB]) / (A [BA] - A [BB])) * (BA - BB)]$$

If the data in that position is equal to the value of data sought (X), then the search is stopped and declared successful [10]. Whereas if the value of data in the position of suspected is not equal to the value of data sought (X) [2] [10], then:

1. For ascending data, if the value of the data sought (X) is smaller than the data value in the suspected position, then the search is continued by changing the upper boundary of the search area. Whereas

if the data value sought (X) is greater than the value of data in the suspected position, then the search is continued by changing the lower boundary of the search area.

For descending data, if the value of the data sought (X) is greater than the value of the data in the suspected position, then the search is continued by changing the upper boundary of the search area. Whereas if the data value sought (X) is smaller than the data value in the suspected position, then the search is continued by changing the lower boundary of the search area.

3. Result and Discussion

Experiment process comparision of linear, binary and interpolation algorithm are using random number data, and the experiment perform to each algorithm, for the value used as follows

Data = 10, 8,5,15,19,35,4,2, 65, 13, 17, 32

For the first test searching process are using linear search algorithm, the result as below:

Data: 10, 8, 5, 15, 19, 35, 4, 2, 65, 13, 17, 32

X   = Data searched

I   = the position of the data compared against X

N   = Amount of data

F   = variable Boolean

A (I)   = Data with the 1st position

X = 4, I = 1, N = 12, F = False, A (1) = 10
While (1 <= 12) And Not (False) -> True, Search resumed.
If (4 = 10) -> False
I = I + 1 = 1 + 1 = 2

X = 4, I = 2, N = 12, F = False, A (2) = 8
While (2 <= 12) And Not (False) -> True, Search resumed
If (4 = 8) -> False
I = I + 1 = 2 + 1 = 3

X = 4, I = 3, N = 12, F = False, A (3) = 5
While (3 <= 12) And Not (False) -> True, Search resumed
If (4 = 5) -> False
I = I + 1 = 3 + 1 = 4

X = 4, I = 7, N = 12, F = False, A (7) = 4
While (7 <= 12) And Not (False) -> True, Search resumed.
If (4 = 4) -> True
Found = True

X = 4, I = 7, N = 12, F = True, A (7) = 4
While (7 <= 12) And Not (True) -> False, Search not resumed.

F = True
Data '4' Found!

The process of searching data with the pseudo-linear process has been performed, following test search data with the binary algorithm, the result as below:

Data: 2, 4, 5, 8, 10, 13, 15, 17, 19, 32, 35, 65

X   = Data searched

N   = Amount of data

Left   = the left variable value of the binary algorithm
Right   = the right variable value of the binary algorithm
Middle   = the middle variable value of the binary algorithm
Found   = variable Boolean
A (Mid) = Data with position to - < middle value >

X = 4, Left = 1, Right = 12, Middle = 0, Found = False
While (1 <= 12) and Not (False) -> True,
Middle = (Left + Right) \ 2 = (1 + 12) \ 2 = 6
If (4 = 13) -> False
(4 < 13) -> True
Right = Middle - 1 = 6 - 1 = 5

X = 4, Left = 1, Right = 5, Middle = 6, Found = False, A (6) = 13
While (1 <= 5) And Not (False) -> True, Search resumed.
Middle = (Left + Right) \ 2 = (1 + 5) \ 2 = 3
If (4 = 5) -> False
(4 < 5) -> True
Right = Middle - 1 = 3 - 1 = 2

X = 4, Left = 1, Right = 2, Middle = 3, Found = False, A (3) = 5
While (1 <= 2) And Not (False) -> True, Search resumed.
Middle = (Left + Right) \ 2 = (1 + 2) \ 2 = 1
If (4 = 2) -> False
(4 < 2) -> False
Left = Middle + 1 = 1 + 1 = 2

X = 4, Left = 2, Right = 2, Middle = 1, Found = False, A (1) = 2
While (2 <= 2) And Not (False) -> True, Search resumed.
Middle = (Left + Right) \ 2 = (2 + 2) \ 2 = 2
If (4 = 4) -> True
Found = True

X = 4, Left = 2, Right = 2, Middle = 2, Found = True, A (2) = 4
While (2 <= 2) And Not (True) -> False, Search not resumed.

Found = True
Data '4' Found!

The last using with interpolation search algorithm as follows:
Data: 2, 4, 5, 8, 10, 13, 15, 17, 19, 32, 35, 65
X   = Data searched
N   = Amount of Data
BB   = Lower Boundary Value
BA   = Upper Limit Value
Pos   = the position of the data being compared
CekPos = CekPos Variable Value
Found = Variable Boolean
A (Pos) = Data with position to - <pos value>

X = 4, BB = 1, BA = 12, Pos = 0, CekPos = 0, Found = False

While (1 <= 12) And Not (False) -> True, Search resumed.
Temp = BB + ((X - A (BB)) / (A (BA) - A (BB))) * (BA - BB)
Temp = 1 + ((4 - 2) / (65 - 2)) * (12 - 1) = 1.349206
Pos = Int (Temp) = Int (1.349206) = 1
If (1.349206 - 1 > 0) -> True
Pos = Pos + 1 = 1 + 1 = 2
If (0 = 2) -> False
(4 = 4) -> True
Found = True

X = 4, BB = 1, BA = 12, Pos = 2, CekPos = 0, Found = True, A (2) = 4
While (1 <= 12) And Not (True) -> False, Search not resumed

Found = True
Data '4' found!

The last process of the interpolation search has been completed, from the above three search algorithm process using same length of data and same value, based on the pseudo process of each algorithm and by re-testing for longer lengths get the result as follows:

Table 1. Result Searching

No	Length Number	Algorithm		
		Linear (Millisecond)	Binary (Millisecond)	Interpolation (Millisecond)
1	500	3.35	3.03	2.54
2	1000	5.12	6.19	4.51
3	2000	7.54	7.41	6.42

Based on the above process with a search of different lengths results obtained under the interpolation search algorithm more quickly than linear and binary algorithms
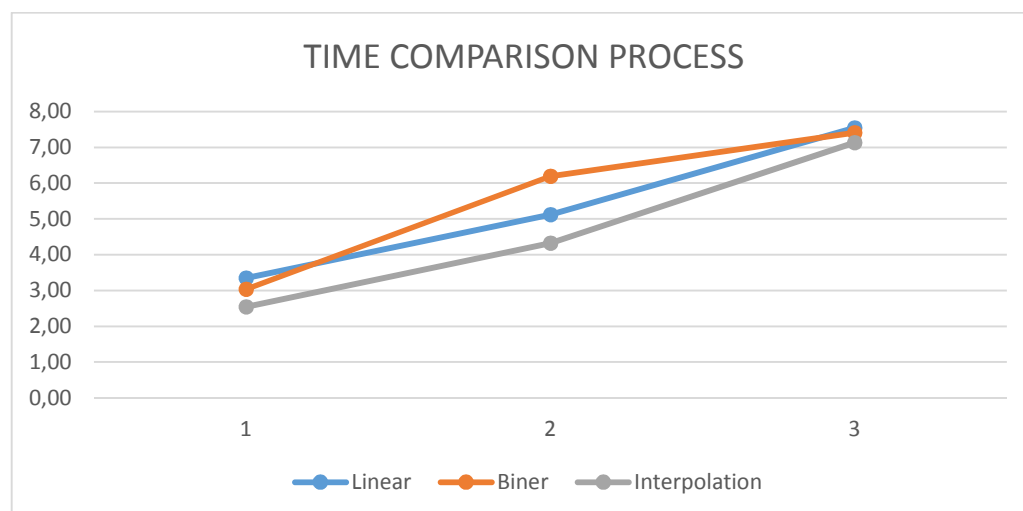

Figure 1. Time Comparison

Experiment was perform using Python programming because have a stable memory and large memory cache so that the experiment process can be done as objective as possible.

4. Conclusion

The comparison of linear, binary and interpolation search algorithms is made by testing with data lengths of 500, 1000 and 2000, based on the authors experiment it appears that the interpolation search algorithm is faster than the other two algorithms, the results of this test are not absolute for all cases that can be implemented, but this test using pseudo code approach process with random data that the author did and re-examined until a couple of times by using a program created by the Python language, the results that interpolation search algorithm faster

References

[1] S. Pushpa and P. Vinod, "Binary Search Tree Balancing Methods: A Critical Study," *International Journal of Computer Science and Network Security,* vol. 7, no. 8, pp. 237-243, 2007.

[2] J. z. Li and H. K. Wong, "Batched Interpolation Searching on databases," in *IEEE Third International Conference on Data Engineering*, Los Angeles, CA, USA, USA, 1987.

[3] S. Kareem B, "Matrix Search Algorithm Using Binary Search Trees," *Research & Reviews: Journal of Engineering and Technology,* vol. 5, no. 2, pp. 1-6, 2016.

[4] N. Arora, G. Bhasin and N. Sharma, "Two way Linear Search Algorithm," *International Journal of Computer Applications,* vol. 107, no. 21, pp. 6-8, 2014.

[5] B. G. Balogun and J. S. Sadiku, "Simulating Binary Search Technique Using Different Sorting Algorithms," *International Journal of Applied Science and Technology ,* vol. 3, no. 6, pp. 67-75, 2013.

[6] K. K. Pandey and N. Pradhan, "A Comparison and Selection on Basic Type of Searching Algorithm in Data Structure," *International Journal of Computer Science and Mobile Computing,* vol. 3, no. 7, pp. 751-758, 2014.

[7] V. P.Parmar and CK Kumbharana, "Comparing Linear Search and Binary Search Algorithms to Search an Element from a Linear List Implemented through Static Array, Dynamic Array and Linked List," *International Journal of Computer Applications,* vol. 121, no. 3, pp. 13-17, 2015.

[8] A. Mehta, A. Saxena, J. Patel and A. Thanna, "A REVIEW ON COMPARISION OF BINARY SEARCH AND LINEAR SEARCH," *International Journal of Engineering Sciences & Management Research,* vol. 2, no. 10, pp. 85-89, 2015.

[9] Y. Perl and L. Gabriel, "Arithmetic interpolation search for alphabet tables," *IEEE Transactions on Computers,* vol. 41, no. 4, pp. 493 - 499, 1992.

[10] G. MARSAGLIA and B. NARASIMHAN , "Simulating Interpolation Search," *Computers Mathematics Application,* vol. 26, no. 8, pp. 31-42, 1993.