Analysis of Linear and Binary Search Algorithms

Sapinder  
Assistant Professor  
BKSBD Girls College Amritsar

Ritu  
IGNOU Study Centre  
Amritsar

Arvinder Singh  
Student Khalsa College  
Amritsar

Heera Lal Singh  
Lecturer, SGHB Sen. Sec School  
Tarn Taran

ABSTRACT

The current age is the age of science and technology. In these days the research is focusing on new development and analyzing or comparing the existing technologies. Software engineering is one of the dominant branches of Computer Science and engineering that is associated with the development and analysis of software process and product. One of the emerging areas in the field of Software engineering is the study of software metrics. By using the concept of software metrics one is able to measure the effectiveness of code without its execution. The objective of this study is to analyze the different software metrics like for linear and binary searching algorithms available in commercial market. We have developed a code in visual basic to analyze the difference metrics of linear and binary search algorithms.

General Terms

Software metrics, linear search, binary search.

Keywords

Static metrics, Complexity, Execution Time, LOC, Operands, Operators etc.

1. INTRODUCTION

Software metric is a field of software engineering that is associated with diverse measurements of computer software and its developments. Software metrics [1] [2] [3] is one of the important tools for analyzing the software product in an effective way. In other words software metrics are measures that enable software developers and software analyst to gain insight into the efficiency of the software process and projects that are conducted using the process as framework. Software metrics measures different aspects of software complexity and therefore play an important role in analyzing and improving software quality [3]. With the help of software metric we are able to understand the software product in an effective way. We apply some software logic or mathematical technique to software process or product to supply or improve engineering and management information.

Software metric is a field of software engineering that is associated with diverse measurements of computer software and its development. According to Tom DeMarco that “You cannot control what you cannot measure”. With the help of software metric we are able to measure some property of software or its component. Computer science researchers are putting their all efforts in measuring quantitative information from software component. Software metric [4] [5] are helpful in improving the quality of software, planning the budget, its cost estimation etc. with the help of software metric we are able to understand the software product in an effective way. We apply some software logical of mathematical technique to software process or product to supply or improve engineering and management information.

2. OBJECTIVE OF STUDY

The objective of this paper is to compare and contrast the various sorting and searching algorithms like bubble sort, Selection Sort, linear search and binary search etc. in terms of various Halstead metrics. For analyzing various sorting algorithms we will develop graphical user interface in Visual Basic 6.0 under Microsoft Windows platform. The various sorting algorithms will be compared in terms of their execution time, lines of code, number of operators and operands used, program volume, program length, control density, program vocabulary etc.

3. ANALYSIS

In the research cycle of software metrics starts in 1970, it was Wolverton [6] who performs a research on production ratio of the programmer by using the concept of LOC i.e. line of code. According to Somerville the metric can be classified into two categories i.e. control metric and predictive metric. Predictive metric are normally associated with software product. With the help of predictive metric [7] we are able to determine both static as well as dynamic characteristics of the software. There are two major types of predictive metrics i.e. Static and Dynamic Metrics.

One can also search an item from an array by using two major techniques known as linear search and binary search. Here linear search is simple to implement thought takes larger time while searching an item, where as binary search algorithm search an item with smaller time as compared to linear search. The algorithmic working of linear and binary search is given as below:
Working of linear search:

Read item
For I = 0 To N-1
    If Item=Arr[i] Then
        flag = 1
        print “Item found”
    Else
        flag = 0
    End If
I=I+1
End For
If flag = 0 Then
    MsgBox “ITEM NOT FOUND”
    GoTo fail
End If

The various software metrics analyzed during linear searching an item value 6 search are as given in the following table:

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOC</td>
<td>11</td>
</tr>
<tr>
<td>n1</td>
<td>1</td>
</tr>
<tr>
<td>n2</td>
<td>3</td>
</tr>
<tr>
<td>N1</td>
<td>5</td>
</tr>
<tr>
<td>N2</td>
<td>9</td>
</tr>
<tr>
<td>Execution Time</td>
<td>2.625000</td>
</tr>
<tr>
<td>Program Length</td>
<td>14</td>
</tr>
<tr>
<td>Program Vocabulary</td>
<td>4</td>
</tr>
<tr>
<td>Program Volume</td>
<td>19.4081210556785</td>
</tr>
</tbody>
</table>

The graphical representation of the above software metric is as follow:

![Software Metrics for Linear Search](image)

Figure1: Software Metrics for Linear Search

Working of binary search:

flag = 1
flag = 0
lower = 0
upper = 8
While (lower <= upper)
    md = (lower + upper) / 2
    If item = arr[md] Then
        upper = md - 1
    Else
        lower = md + 1
    End If
End While
If upper = lower Then
    MsgBox “ITEM NOT FOUND”
    GoTo fail
End If
Locate the item using the mid-point of the array and narrow down the search range.
lower = md + 1
End If
Wend
fail:
If flag = 0 Then
MsgBox "Item not found"
Else
MsgBox "Item found"
End If

The various software metrics analyzed during binary search for searching an item value 6 search are as given in the following table:

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOC</td>
<td>11</td>
</tr>
<tr>
<td>n1</td>
<td>7</td>
</tr>
<tr>
<td>n2</td>
<td>7</td>
</tr>
<tr>
<td>N1</td>
<td>25</td>
</tr>
<tr>
<td>N2</td>
<td>18</td>
</tr>
<tr>
<td>Execution Time</td>
<td>1.218999</td>
</tr>
</tbody>
</table>

The following chart shows the graphical representation of above computed metrics for binary search.

**Figure 2:** Software Metrics for Binary Search

4. CONCLUSION

From the above analysis it is clear that binary search algorithm has more line of code, program volume, program vocabulary, program length but still takes lesser execution time in more number of cases as compare to linear search while searching an item. No doubt binary search is difficult to implement since it also requires sorting of number before searching, but give more optimized result as compare to linear search.

5. REFERENCES


