Analysis of Software Metrics for Bubble Sort and Selection Sort

Amrit Dhillon
M.Tech. (CSE) Student
LPU, Phagwara.

Amrinder Singh
Student, Computer Dept
PTU Study Centre.

ABSTRACT

We are living the era 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 branch of Computer Science and engineering that is associated with the development and analysis of software process and product. One of the emerging area 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 for most commonly used sorting techniques like bubble sort and selection sort. We will develop and analyze the above said sorting algorithms to analyze the significance of different performance metrics on Software/Simulator developed in visual basic.

Keywords

component; XML Schema, XML Schema Resemblance, XML Schema Matching, Similarity Measure.

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.

Sorting is the way to arrange the data in either ascending or in descending order. There are various sorting algorithms available in commercial market. The objective of my study is to measure the various static metrics for some sorting and searching algorithms.

Bubble sort is one of the important and most frequently used sorting algorithms.

```
For I = 0 To 8
    For j = 0 To 8 - I
      If Val(Txt1(j).Text) > Val(Txt1(j + 1).Text) Then
        temp = Val(Txt1(j).Text)
        Txt1(j).Text = Val(Txt1(j + 1).Text)
        Txt1(j + 1).Text = temp
      End If
    Next j
    Next I
```

The various software metrics that are analyzed while sorting an array of 10 elements is as given below:

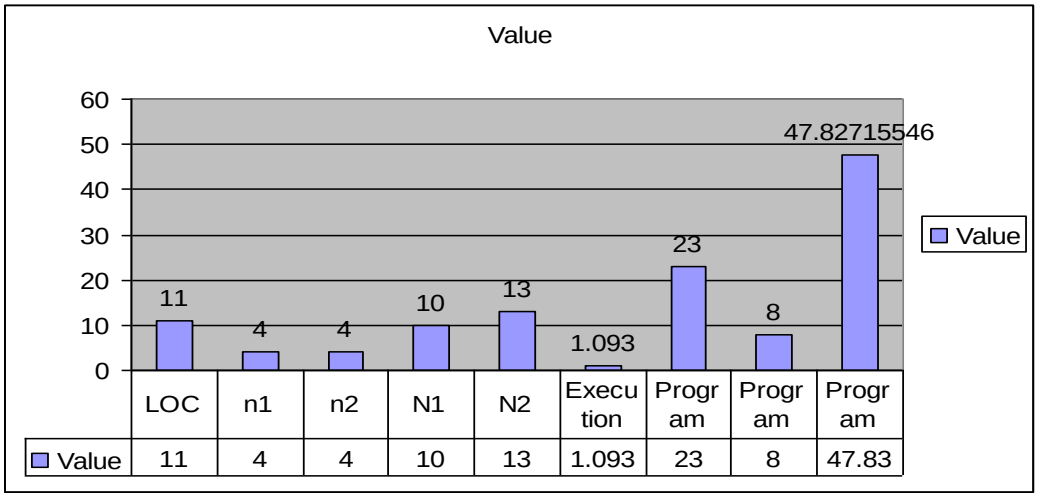
Test data: 1, 3, 33, 2, 3 52,23,52,23,123

After sorting by using bubble sort the data come in the order as 1,2,3,3,23,23,33,52,52,123.

The various software metrics analyzed during bubble sort are as given in the following table:

Metric	Value
LOC	11
n1	4
n2	4
N1	10
N2	13
Execution Time	1093000.00000076 (micro seconds)
Program Length	23
Program Vacublory	8
Control Density	9.09090909090909E-02
Program Volume	47.8271554586362

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



Metric's analysis for Bubble sort

The logarithmic relationship computer between different computer attributes is as given below:

y = 8.9321Ln(x) + 0.8415

selection sort is other important sorting algorithm that is used to sort an array or number of items in either ascending or descending order.

```
Dim I As Integer
  Dim j As Integer
  For I = 0 To 8
    For j = I + 1 To 8
      If Val(Txt1(I).Text) > Val(Txt1(j).Text) Then
        temp = Val(Txt1(I).Text)
        Txt1(I).Text = Val(Txt1(j).Text)
        Txt1(j).Text = temp
      End If
    Next j
    Next I
```

The same set of data when was analyzed by using selection sorting algorithms the following values are computed against various software metrics computed above.

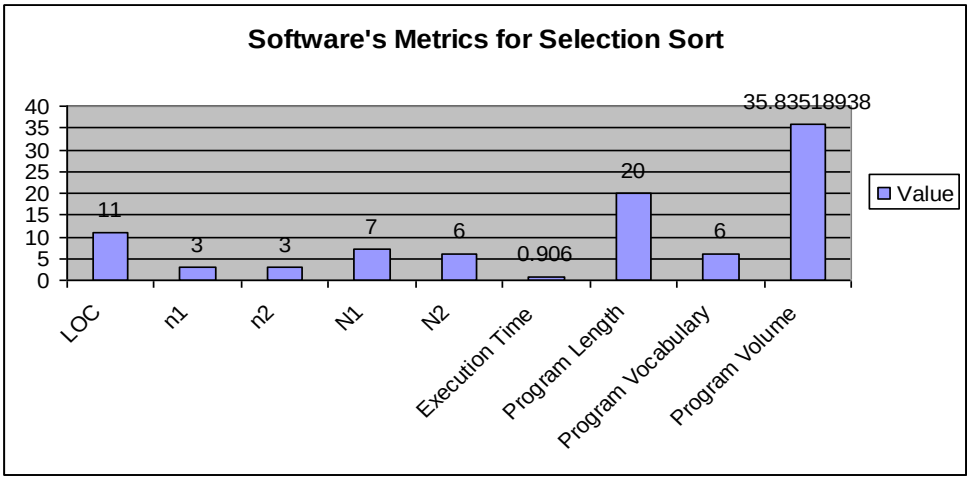
Test data: 1, 3, 33, 2, 3 52,23,52,23,123

After sorting by using bubble sort the data come in the order as 1,2,3,3,23,23,33,52,52,123.

The various software metrics analyzed during bubble sort are as given in the following table:

Metric	Value
LOC	11
n1	3
n2	3
N1	7
N2	6
Execution Time	906000.000002678 (micro seconds)
Program Length	20
Program Vocabulary	6
Control Density	9.09090909090909E-02
Program Volume	35.8351893845611

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



Software metrics of selection sort

4. CONCLUSION

In sorting it is clear from above tables and graphical representation that although selection sort takes almost same line of code as bubble sort, but it has lesser value of program length, program vocabulary, and program volume, beside this the study shows that selection sort takes lesser time while sorting as compare to bubble sort. 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

[1] H F Li, W K Cheung "An Empirical Study of Software Metrics" Software Engineering IEEE Transactions on (1987) Volume: SE-13, Issue: 6, Pages: 697-708

[2] N E Fenton "Software Metrics" Conference Proceedings of on the future of Software engineering ICSE 00(2000) Volume: 8, Issue: 2, Publisher: ACM Press

[3] manik Sharma, Gurdev Singh, "Analysis of Static and Dynamic Metrics for Productivity and Time Complexity", IJCA, *Volume 30– No.1, September 2011*

[4] Sharma, Manik, et al. "A comparative study of static object oriented metrics." *International Journal of Advancements in Technology* 3.1 (2012): 25-34.

[5] S.R Chidamber and C.F. Kemerer, "A Metrics Suite for Object Oriented Design", IEEE Transactions on Software Engineering, Vol. 20 No. 6

[6] Tu Honglei, Sun wei, Zhang Yanan, "The Research of Software metric and software complexity metrics" International Forum on Computer Science Technology and Applications (2009) Publisher: IEEE, Pages: 131-136

[7]. Somerville "Software Engineering" 6th Edition, Editor: Addison Wesley.

[8] Sharma, Manik, and Gurvinder Singh. "Predictive Metric-A Comparative Study." *International Journal of Computer Science and Technology (IJCST)* 2 (2011).

[9] Sharma, Manik, and Gurdev Singh. "Static and Dynamic metrics-A Comparative Analysis." *Emerging Trends in Computing and Information Technology* (2011).