Study of Various Software Metrics for Insertion Sort

Amit Puri
Assistant Professor Department of Computer Applications
GIMET, Verka, Amritsar

ABSTRACT
Algorithm is one of the most important components of programming and software engineering. From the past years we are trying to analyze the different type of algorithms. Another important term in the field of software engineering that is used to analyze the algorithm is Software metrics. Software metrics have become an integral part of software development and are used during every phase of the software development life cycle. It is one of the vital tools that can be used to find significant estimates for software products and directs us in taking managerial and technical decisions. Research in the area of software metrics tends to focus predominantly on static metrics that are obtained by static analysis of the software artifact. In this study I will try to analyze the working of one of the important sorting algorithm called insertion sorting. In this study I will implement the insertion sort algorithm in C language and then try to find different software metrics for the same.

Keywords
Sorting, Insertion Sort, Metrics, Complexity.

1. INTRODUCTION
In the field of software engineering the term software metric [1] is one of the important indicators of a programming segment of section. Software metrics plays an important role in understanding the important concepts in the field of software engineering. The name software metric [1][2] is associated with diverse measurements of computer software and its development.

It helps us in measuring [4][5] the performance of various features of the software. With the help of software metric once can measure some property of software or its component. Computer science researchers are putting all their efforts in measuring quantitative information from software component. Software metric [3] are helpful in improving the quality of software, planning the budget, its cost estimation etc.

In this paper I will try to perform metric analysis on one of the important sorting algorithm called insertion sort. Insertion sort algorithm is used to arrange the data in either ascending or in descending mode. In this paper the focus will be given on analyzing the static metrics.

2. INSERTION SORT
Sorting is one of the fundamental operation of data structure. It is mainly used to arrange the data in either ascending or in descending order. Insertion sort is one of the important sorting algorithms that is mainly used to sort an array of numbers or strings. It is found that[3] unlike many sorting algorithms with quadratic complexity, it is actually applied in practice for sorting small arrays of data. An insertion sort algorithm considers the elements one at a time, inserting each in its suitable place among those already considered. The study [6] has shown that insertion sort algorithm is easy to implement and computed. The worst case and average complexity of insertion sort is as given below:
Worst Case: $O(n^2)$
Averages Case: $O(n^2)$
The practical implementation of insertion sort in ‘C’ language is given as below:

/* Program for Sorting an array using Insertion Sort*/
#include<stdio.h>
#include<conio.h>
void main()
{
    int i,j,s,temp,a[20];
    clrscr();
    printf("Enter Number of elements to Sort ");
    scanf("%d",&s);
    printf("Enter %d elements: ",s);
    for(i=0;i<s;i++)
        scanf("%d",&a[i]);
    for(i=1;i<s;i++)
    {
        temp=a[i];
        j=i-1;
        while((temp<a[j])&&(j>=0)){
            a[j+1]=a[j];
            j=j-1;
        }
        a[j+1]=temp;
    }
    printf("Result After Insertion Sort is: ");
    for(i=0;i<s;i++)
        printf(" %d",a[i]);
    return 0;
}

text should be in two 8.45 cm (3.33”) columns with a .83 cm (.33”) gutter.
3. ANALYSIS OF INSERTION SORT

The main focus of my study is to analyze the different metrics of the insertion sort algorithm. For analyzing the same I have implemented the logic of insertion sort algorithm in ‘C’ language for the simplicity. After implementing the code the following software metrics are computed for analysis.

<table>
<thead>
<tr>
<th>S.No.</th>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>n2</td>
<td>13</td>
</tr>
<tr>
<td>2</td>
<td>n1</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>N2</td>
<td>60</td>
</tr>
<tr>
<td>4</td>
<td>N1</td>
<td>31</td>
</tr>
<tr>
<td>5</td>
<td>LOC</td>
<td>25</td>
</tr>
<tr>
<td>6</td>
<td>NSC</td>
<td>20</td>
</tr>
<tr>
<td>7</td>
<td>NTM</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>SLOC</td>
<td>25</td>
</tr>
</tbody>
</table>

The graphical representation of the above computer metrics is given as below. I have also tried to find the trend of the computed metric and value in an exponential form.

The following table shows the analysis of insertion sort algorithm in the form of different metrics computed for file as well as for method.

The various computer metrics of Insertion Sort Program implemented in C programming is given as below. The focus is given on computing various Halstead metrics with various size and complexity metrics.

<table>
<thead>
<tr>
<th>S.No.</th>
<th>Name of Metric</th>
<th>Description of Metrics</th>
<th>Metric Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>V(G)</td>
<td>Cyclometric Complexity</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>V'(G)</td>
<td>Extended Cyclometric Complexity</td>
<td>10.10</td>
</tr>
<tr>
<td>3</td>
<td>Depth</td>
<td>Greatest Depth of Method</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>B</td>
<td>Halsted Bug Prediction</td>
<td>0.12</td>
</tr>
<tr>
<td>5</td>
<td>V</td>
<td>Halstead Program Volume</td>
<td>413</td>
</tr>
<tr>
<td>6</td>
<td>Branch</td>
<td>Number of Branching nodes</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>Control</td>
<td>Number of Control Statement</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>Exec</td>
<td>Number of Executable Statements</td>
<td>14</td>
</tr>
<tr>
<td>9</td>
<td>OC</td>
<td>Operational Complexity</td>
<td>50.50</td>
</tr>
</tbody>
</table>
After computing the various metrics, I have tried to create the graphical representation and logarithmic trend of above computed method metrics of insertion sorting algorithm. The following graph show how various computed metrics are interrelated to one another. I have also provided the logarithmic trend line of various computed metric.

![Logarithmic Trend of Method Metrics](image)

**Figure 2: Logarithmic Trend of Method Metrics**

### 4. CONCLUSION

From the above analysis it is clear the extended cyclomatic complexity for the insertion sort algorithm is almost double the cyclomatic complexity. Further analysis also shows that insertion sort algorithm is free from branching statement where as it has significant control statement as compare to executable statements. The operation complexity of the algorithm is almost 1/8th of program volume.

### 5. ACKNOWLEDGMENTS

My special thanks go to my family and my teachers, last but not least to GOD.

### 6. REFERENCES


