**ABSTRACT**

In Computer Science, **Merge Sort** is an efficient, general-purpose, and comparison-based sorting algorithm. Most implementations produce a stable sort, which means that the order of equal elements is the same in the input and output. Merge sort is a divide-and-conquer algorithm that was invented by John von Neumann in 1945.

Given a list of array elements, we want to rearrange them in ascending order. Sorting

algorithms such as the Bubble, Insertion and Selection Sort all have a quadratic time complexity

that limits their use when the number of elements is very big. But in Merge Sort, a divide-and conquer algorithm is used to sort an N element array. We evaluate the **O (n log n )** time complexity of merge sort theoretically and empirically. Our results show a large improvement in

efficiency over other algorithms.

**TABLE OF CONTENTS**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  | Acknowledgement  Abstract  Table of contents  List of figures | i  ii  iii  iv | | **1** | **INTRODUCTION**  1.1 Introduction to Computer graphics  1.2 Introduction to OpenGL  1.3 About Project | 1  1  3  7 | | **2** | **REQUIREMENTS AND SPECIFICATIONS**  2.1 Hardware Requirements  2.2 Software Requirements | 8  8  8 | | **3** | **SYSTEM DESIGN**  3.1 User Inputs  3.2 Block Diagrams | 9  9  10 | | **4** | **IMPLEMENTATION**  4.1 Pseudocode  4.2 Main Function  4.3 Function for Merge Sort  4.4 Flow chart  4.5 Functions used | 11  11  11  13  14  15 | | **5** | **TESTING**  5.1 Introduction to Testing  5.2 Stages in the implementation of Testing | 16  16  16 | | **6** | **SNAPSHOTS** | 17 | |  | **CONCLUSION**  **REFERNCES** |  | |  |  |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURE NUMBER** | **DESCRIPTION** | **PAGE NUMBER** |
| Figure 1.1 | Components of Graphics Architecture and their  Working. | 02 |
| Figure 1.2 | OpenGL Library Organization. | 03 |
| Figure 1.3 | OpenGL Order of Operations. | 05 |
| Figure 3.1 | Working of Merge sort Algorithm . | 10 |
| Figure 4.1 | Flow Chart for implementation of Merge Sort  Algorithm. | 14 |
| Figure 6.1 | Entering the array elements. | 17 |
| Figure 6.2 | Randomly placed array elements. | 17 |
| Figure 6.3 | Two halves of sorted array. | 18 |
| Figure 6.4 | Sorting of array by dividing them. | 18 |
| Figure 6.5 | Fully Sorted array elements. | 19 |