

A Comparative Study of Sorting and Searching Algorithms

Ms ROOPA K¹, Ms RESHMA J²

¹IV sem M.Tech, Department of Computer Science and Engineering, BNM Institute of Technology, Bengaluru, India

²Assistant Professor, Department of Computer Science and Engineering, BNM Institute of Technology, Bengaluru, India

ABSTRACT - The modern era is the age of expertise and focuses on new development along with the comparing of existing technologies. Here, a comparative study aims to come up with the most efficient sorting algorithms or the techniques used. Sorting is an important data structure operation for managing data. Sorting algorithms and searching techniques are both distinct. Searching is based on searching the element for the given list, where in Sorting is to arrange the given list in a particular order which can be in ascending or descending order.

Most of these data are being compared, sorted only for a portion and the piece of data which is actually used for determining sorted order is known as the Key. The methodology used is to evaluate the performance using CPU time and memory space. Sorting algorithms and searching technique can operate both on sorted and unsorted list of elements. The items may be stored individually as records in a database or may be elements of a search space defined by a procedure.

There are different types of algorithms and techniques for performing different tasks and as well as same tasks, with each having its own advantages and disadvantages depending on the type of data structure. An analysis is being carried out on various sorting algorithms and searching techniques on parameters like space and time complexity. Based on the analysis, a comparative study is being made so that the user can choose the type of technique used based on the requirement.

Key Words: Quick sort, Selection sort, Bubble sort, Binary search, Comparisons.

1. INTRODUCTION

In the latest technology, the system can store large and complex amounts of item or data from which the user can access this item or data either individually or in bulk. Consider for an example, if a contact number provided in the telephone directory has to be searched. Then, the data has to be sorted and searched.

Searching is a process of identifying a specific item or data and helps in recovery of a particular item or a data from the given set of items in a list. Where in based upon this, the answer is obtained like the item which is to be recovered is present or not in the given list. As a result, it requires determined sorting techniques and searching

algorithm to perform the process of searching and fetching the key from the given list within less time.

A comparative study of sorting techniques and searching algorithm based upon time and space complexity is discussed. Different sorting techniques and searching algorithms along with the implementation are dependent upon situation. Usually, sorting techniques depends mainly with two parameters, in which, first parameter tells the time taken for execution the data and second parameter tells about the speed or space taken for execution of the data. Some of the different sorting techniques are as follows:

- Quick sort
- Selection sort
- Bubble sort
- Insertion sort
- Merge sort
- Heap sort

1.1 Quick Sort

Quick sort as the name suggests helps in sorting list of items very quickly with less space and is based upon the Divide and Conquer principle also known as a Partition Exchange sort[6]. This type of sorting is known as one of the best sorting algorithms as it has efficiency and can handle complex and vast lists of item without using extra storage in it. It takes the following steps for processing of an item or data for the given list:

- Step 1: Pick the element from the given list and use as a pivot.
- Step 2: Partition these elements into two sub-lists.
 - Elements less than pivot.
 - Elements greater than pivot.
- Step 3: Quick sort two sub-lists.
- Step 4: Repeat the process until the entire lists of item are sorted.
- Step 5: Return result.

Below figure 1 shows a simple representation of Quick sort [7] technique considering the pivot elements.

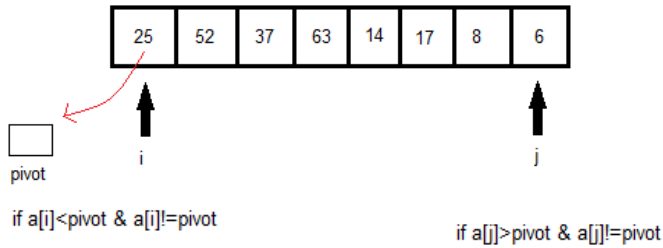


Fig -1: A Simple Representation of a Quick Sort

The advantages and disadvantages of Quick Sort are as follows[8]:

Advantages:

- This is one of the best sorting algorithms.
- It can handle large items for the given lists.
- Does not require extra storage.

Disadvantages:

- The worst-case performance is same as to bubble, selection and insertion sort.
- In case the given lists are already sorted, then bubble sort is more efficient compared to quick sort.
- For sorting the integers of given items, radix sort plays an important role as compared to quick sort.

1.2 Selection Sort

One of the simplest techniques used is the selection sort which works for the given lists of item or data each time by selecting one item at a time and orders along with placing it in a correct position in sequence. This does not require extra storage space to hold the lists[11]. For selecting the item randomly this technique is helpful.

The following steps are used for processing the elements in given lists of items.

- Step 1: First finds the smallest element for the given lists of item or data.
- Step 2: Replaces that item or data in the first position.
- Step 3: Next, again finds the smallest element among the lists.
- Step 4: Get replaced in second position.
- Step 5: Same procedure is followed unless the elements are sorted.
- Step 6: Returns result.

Original Array	After 1st pass	After 2nd pass	After 3rd pass	After 4th pass	After 5th pass
3	1	1	1	1	1
6	6	3	3	3	3
8	8	8	8	4	4
4	4	4	6	5	5
5	5	5	5	6	6
				8	8

Fig -2: Selection Sort for Different Passes

Above figure 2 shows the number of passes taken for the given inputs[7]. The advantages and disadvantages of Selection Sort are as follows[8]:

Advantages:

- Before sorting the given lists, the ordering of items or datas can be initialized.
- This works even for the smaller lists of items or datas.
- Need no extra storage for the original lists of items.

Disadvantages:

- For the large set of items or datas this sorting results in poor efficiency.
- Requires an N-squared number steps for sorting items for the given lists.
- Compared to selection sort, the quick sort is most efficient.

1.3 Bubble Sort

The bubble sort works on the basis for given lists by swapping adjacent item or data in order, until the entire lists of item or data are obtained in a sequence based upon their key values. This technique is popular and easy for execution of data in which it does not require extra storage space. This takes the following steps as shown below:

- Step 1: Consider the set of elements which can be of any item or data for the given lists.
- Step 2: Swap the first two items or data for the given lists adjacently using one by one pairs.
- Step 3: Repeat the process for all the items or data in the given list.
- Step 4: Return result for the same.

And figure 3 represents the simple step by step swapping of numbers for the given inputs[7].

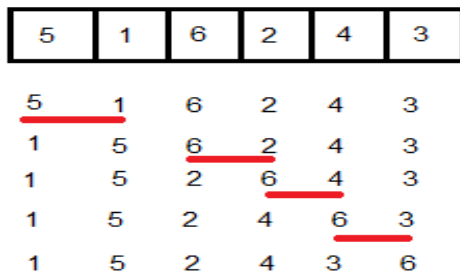


Fig -3: Bubble Sort Technique

The advantages and disadvantages of Bubble Sort are as follows[8]:

Advantages:

- It is one of the popular and easily implemented algorithms.
- Without using the extra storage, the items can be swapped easily.
- Requires minimum space for given lists.

Disadvantages:

- Requirement of N-squared number steps for sorting items.
- This type of sorting does not deal with large number of items for the given lists.
- Appropriate for academic teaching and cannot be used for real time applications.

As discussed earlier, searching algorithms becomes the most important part of the any programs[9]. Searching can be involved in many ways either to search an entry for the given database for the lists of items or datas, available virtual space, which can be the suitable searching algorithm, based upon time and space density and so on. Entire array lists are sorted in order faster as compared to other algorithms. Some of the algorithms are as follows:

1. Linear Search
2. Binary Search
3. Tree Search
4. Jump Search
5. Interpolation Search
6. Exponential Search
7. Fibonacci Search

Here, the main concentration is on Binary Search algorithm which is the most popular algorithm used for a complex and large database along with records ordered in an associated statistical key format[10]. This uses a recursive technique for searching a specific item from the given lists, resulting in reducing the searching space. The prerequisite for binary search is that the list should be in sorted order. Steps to be followed while performing Binary search are as follows:

- Firstly, it works by dividing the given lists of an array into halves and looks at the middle element of the list.
- In case the key element is smaller than the middle element, then the second half of the given list is ignored and the first half of the list is considered.
- Otherwise, if the key element considered is greater than the middle value, then the first half of the given list gets ignored and only the second half of the list is considered.
- Repeat with the left over half until the key element is found.
- In case the key element is found, it is returned else it returns -1 if the key element is not found.

The advantages and disadvantages of Binary Search are as follows[8]:

Advantages:

- This is one of the simplest algorithms.
- Checking of each item or data for the given list of array is faster.
- Takes $\log_2(N)$ $\log_2(N)$ evaluations for average and worst-case.

Disadvantages:

- The binary search is most complex search when compared to other searches.
- In a given array list, if the items is in sorted sequence only then this searching mechanism works.
- If the searching technique does not support random access of items for the given lists then it leads to a lot of efficiency.

2. RELATED WORKS

An era of technical knowledge is being increasing rapidly for the analysis and comparative of different types algorithms used for searching any type of complex or large data's given for the lists of items or data's. Focuses on expansion of different searching techniques such as binary or linear searches related to analysis and comparisons that are made for the given lists of items or data's so as to find out which searching technique is more efficient in resulting determination of time[3].

Searching refers to a process of inspecting whether the required item or data for the given lists is present or not which can be done using the key value. Some of the different types of techniques are sequential search, binary search, linear search, tree search and so on. Based upon these searching techniques, a single record one at a time can be fetched and also recovered from the storages that are available either internally or externally within a minor amount of time for the user.

The binary and linear searches are the two techniques considered in a given lists of items or data's for searching essentials[3]. In case of binary search instead of checking each and every item one by one can be sorted in order and saves time. Whereas in case of linear search increases the time for searching a particular item or data. A comparison table between binary and linear searches can be illustrated as follows[12]:

Basis of Comparison	Binary Search	Linear Search
Time Complexity	$O(\log_2 N)$	$O(N)$
Best case time	Center Element $O(1)$	First Element $O(1)$
Prerequisite for an array	Array must be in sorted order	Not required
Worst case for N number of elements	Can conclude after only $\log_2 N$ comparisons	N comparisons are required
Can be implemented on	Cannot be directly implemented on linked list	Array and Linked list
Insert operation	Require processing to insert at its proper place to maintain a sorted list.	Easily inserted at the end of list
Algorithm type	Divide and conquer in nature	Iterative in nature
Usefulness	Somehow tricky algorithm and elements must be arranged in order	Easy to use and no need for any ordered elements
Lines of Code	More	Less

Table 1: Comparison table of Binary Search and Linear Search

This results that when both the searches such as binary and linear searches are used for comparisons of the given set of lists of items or data's. Binary search can be used or designed for the larger and complex lists of items or data's and arranged either in ascending or descending order by taking minor amount of time than linear search which can be used for less significant data's.

Sorting algorithm assists in depth knowing about the definitions of each and every algorithm, work procedures, time and space complexities and so on. In general, is used to compare the difference between the swapping of records, requirement of memory space and estimation made between worst, average and best positions[2].

Here, examines mainly about the bubble, selection and gnome sorts along with some of the most important algorithms such as Divide and Conquer and Dynamic Programming etc. Hence, whenever an inputs are given for these three algorithms estimations of time for each and every algorithm are calculated and results that none of these algorithm can provide an accurate solution. Also, gives information about the algorithm which can be defined as the line by line procedure for explaining exact problem. It holds a set of protocols while processing each and every lines of algorithm.

This specifies that different types of sorting algorithms enclose its own method for solving the worst, average and best positions based upon the inputs along with the space used and gives an idea that more than one algorithms can be used for solving numerous problems[4]. Finally, concludes that when used the similar set of inputs which are been sorted for these algorithms, study shows that the gnome sort calculates the time very rapidly for the given set of inputs when compared along with selection and bubble sorts. Whenever for the given inputs in case if necessary the exchange of inputs can be done using bubble and gnome sorts.

Modern technology has evolved advanced facts for the expansion of required information relating to different types of techniques and algorithms. These techniques and algorithms when used and selected together without making a proper option leads to various problems while processing the data's. Hence, in this survey endeavours[1] in opting for proper techniques and algorithms based upon their time and space complexity. By using different techniques such as bubble sort, selection sort and insertion sort the time and space complexity are carried out and analysis is performed to check for knowing which among these three techniques provides the majority efficiency used, while sorting the data's for the given lists of elements.

The method for dealing with these quantities of items or data's processes sorting of elements that are present in the lists then the searching mechanism takes place for execution of datas. Sorting here refers in filtering of elements for the given lists such as sorting of contact numbers given in telephone directory, sorting of aadhar card numbers in the given data set and so on. For performing a lengthy sorting for the given sequence, a variety of functioning process are required and can come to know which is the suitable algorithm essential for

calculating time and space firmness for the given bubble, insertion and selection sorts[1].

With the help of structural design and then put into practice calculating the estimation of these techniques results in identifying the estimation of time and space taken by these three sorting techniques such as bubble, insertion and selection sorts.

The purpose here is to measure the time estimated for the given linear and binary searching techniques as and when the inputs are selected[5]. Searching generally refers in resulting the required item which has been identified for the provided list of elements. Searching are of two types namely: 1) Internal Searching and 2) External Searching.

- 1) **Internal Searching:** Is the method of searching an data or item for the given set of inputs in Random Access Memory (RAM) is known as Internal Searching.
- 2) **External Searching:** Is the method of searching an data or item for the given set of inputs by not using RAM but externally stored devices is known as External Searching.

Here, gives in depth details of the linear and binary searches and which one among the two searching techniques provides a faster access for the inputs and provides the best time estimation [5]. Some of the advantages and disadvantages of linear and binary searches are also examined. Ultimately, wraps the concepts of searching techniques such as linear and binary searches along with the working procedures and examines the differences between these searching techniques.

3. CONCLUSION

This paper covers some of the popular sorting techniques like quick sort, selection sort and bubble sort. It highlights the advantages and disadvantages of the above sorting algorithms. Binary and linearly search is also explained and compared. It helps in finding out which sorting techniques takes less time and space for the given inputs.

REFERENCES

1. Makinde O.E. et.al, "Performance Evaluation of Sorting Techniques" ,ARPN Journal of Systems and Software, VOL. 3, NO. 5, August 2013.
2. D.R. Aremu et.al, "A Comparative Study of Sorting Algorithms", African Journal of Computing & ICT, Vol 6. No. 5, December 2013 .
3. Ayush Pathak, "Analysis and Comparative Study of Searching Techniques", International Journal of

Engineering Sciences & Research Technology, March, 2015.

4. Jehad Hammad, "A Comparative Study between Various Sorting Algorithms", International Journal of Computer Science and Network Security, VOL.15 No.3, March 2015.
5. Yuvraj Singh Chandrawat et.al, "Analysis and Comparative Study of Searching Techniques", International Journal of Engineering Sciences & Research Technology, October, 2015.
6. <http://faculty.simpson.edu>
7. <https://www.studytonight.com>
8. <http://z-sword.blogspot.in>
9. <https://www.techwalla.com>
10. <https://www.quora.com>
11. <https://www.geeksforgeeks.org>
12. <https://techdifferences.com>

BIOGRAPHIES:



Ms Roopa K is a student from IV sem, M.Tech in Computer Science & Engineering at B N M Institute of Technology, Bengaluru.



Ms. Reshma J is working as Assistant Professor in Department of Computer Science and Engineering at B N M Institute of Technology, Bengaluru. She has over twelve years of teaching experience. She is pursuing her research in the area of Wireless Sensor Networks. She has published articles in International Journals and Conferences.