Lab Assignment - Searching and Sorting Algorithms

Sanjay Reddy Muthyala

Illinois Institute of Technology

Masters In ITM

Chicago, Illinois

February 14, 2022

Describe what is sorting and searching, and why are they essential in a computer science field.

SORTING

The process of arranging elements in a specific sequence. When data is unorganized, ordering it will aid in enhancing readability, understanding, and working with the data in Computer Science.

SEARCHING

Identify whether a specific element is contained in a set of elements. When dealing with many values in computer science, we may need to check for the existence of a specific value and reply accordingly.

Examples of sorting and searching that are necessary for designing software applications.

Example 1

For analyzing students' results, such as determining the topper among a group of students (marks must be sorted in ascending/descending order to determine the clincher), identifying those who have failed in one subject but not another (find the students who have failed in one subject, search for the same student's result in another issue to compare), and so on.

Example 2

When a new book is purchased, the book's details are entered into the library management system. Then, it needs to be sorted in some way (like arranging according to the book's title). Then, when a reader requests a book, it must be found among the available books on the shelf.

Describe three different existing sorting algorithms and two kinds of searching algorithms.

Selection Sort:

To arrange in ascending order, select the first minimal element and place it in the first position, the second minimum element in the group and place it in the second position, the third minimum element in the group, and place it in the third position, and so on.

Selection Sort:

To arrange in ascending order, select the first minimal element and place it in the first position, the second minimum element in the group and place it in the second position, the third minimum element in the group, and place it in the third position, and so on.

Insertion Sort:

Insertion Sort works by inserting a new card into a deck of cards that have already been sorted. Consider one value as a sorted group and the rest as unsorted. Then, from the unsorted group, shift a value to the sorted group. When moving, double-check the value and ensure it's in the right spot. Continue until all of the values have been moved to the sorted group.

Merge Sort:

Merge Sort is based on the concept of Dividing and Conquering. Continue to divide the entire group into two groups until each group only has one value. Then, during the merge, begin merging two groups into one by comparing and organizing the values.

Linear Search:

Linear Search looks for elements in sequential order, starting at the beginning and ending at the conclusion until the component is discovered. If the complete list is examined, but no match is discovered, this indicates that the value does not exist. In addition, it can search through a collection of unsorted values.

Binary Search:

Binary Search divides a bunch of sorted values into two and searches through them. When sorted in ascending order, the items to the left are smaller, and the elements to the right are more

significant for any group component. As a result of this reasoning, the Search is limited to one-half of the results by comparing the value of the search element.

Compare and contrast above choice of sorting and searching algorithms.

SORTING ALGORITHM

The maximum number of comparisons conducted in Selection Sort and Insertion Sort is N*N. The maximum number of comparisons in Merge Sort is N log N, regardless of how the data is spread. As a result, merging Sort would be the best option.

SEARCH ALGORITHM

When the elements are unsorted, linear Search is employed, and binary search is used when the elements are sorted.

When an element is not discovered, Linear Search does a maximum of n checks

When an element is not discovered in Binary Search, a maximum of n/2 checks are performed.

References