Selection Sort

Data Structures Made Easy

DUBLIN CITY UNIVERSITY

Contents

1. ***Definition***

Selection sort is a sorting algorithm that performs as an in-place comparison sort. Selection sort is regarded as inefficient, in terms of performance, for large unsorted lists. Selection sort is known for its simplicity, and in certain circumstances, possess various advantages over more complicated sorting algorithms.

2. ***Implementation***

Selection sort works as follows:

1. Locate the minimum element of the unsorted array.
2. Once located, assign the index of the minimum element into a minimum index variable.
3. Swap the minimum element into the first position within the array.
4. Continually sort the remaining elements of the array, whilst excluding the previously sorted elements.
5. Repeat steps until entire array is sorted.

3. ***Example***

4. ***Functions***

The implementation of selection sort is based on the following functions

- **selection()**
  This function locates the minimum element within the array and swaps it with the element in the first position. The function selects the next element and locates the next minimum element within the array, whilst ignoring the previous sorted elements. The element is swapped with the element of the minimum index position.

  The remaining values are sorted using the same technique until all elements within the array are correctly sorted.

5. ***Pseudocode***

```
function selection ( array arr, size ){

        for index_1 ( 0 to size of array – 1 )
                minimum = index_1

                for index_2 ( index_1 to size of array )
                        minimum = index_2

                swap ( index_1, minimum)
}
```

6. ***Complexity***

The complexity of selection sort is O(N ^ 2). Selection sort possesses the same complexity as bubble sort; however it performs greater than bubble sort.

The computational complexity of selection sort could be improved to O(N log N) by the use of heap sort. Heap sort improves the selection sort algorithm by using an implicit heap data structure, which will increase the speed of locating and removing the next lowest element within the array.

7. ***Advantages of Selection Sort***

Selection sort is one of the most simplistic sorting algorithms to implement. As it is an in-place sorting algorithm, it does not require any additional storage space. This greatly improves the memory usage required by the sorting algorithm.

8. ***Disadvantages of Selection Sort***

Selection sort's computational time complexity contributes to its disadvantages. Due to selection sort having a complexity of O(N ^ 2), it results in a poor performance upon large arrays with unsorted elements. This is usually the reason why this sorting algorithm is rarely used in large applications.

9. ***References***

Swartz F. 2006. *Selection Sort* [Online]. Available from: http://www.leepoint.net/notes-java/data/arrays/31arrayselectionsort.html

Snider M. 2012. *Selection Sort* [Online]. Available from: http://mattsnider.com/selection-sort

Sortware Technologies 2012. *Selection Sort* [Online]. Available from: http://tactics4interview.blogspot.ie/2012/02/selection-sort.html