Rohan Reddy Mandha
1002236945

Correctness of Selection Sort Algorithm

Let us find the smallest element in the unsorted part of the array and swap it with the first element in the part of array

```
    for i=0 to n-1 {

    minimum = i
   for j = i+1 to n {
         if (arr [j] < arr [min]) {
              minimum = j
     }
  }

     if (i=1) {
       swap (ith element , minimum element)

  }
}
```

Correctness:-

The Correctness of the selection Sort algorithm can be proven by using a loop invariant and demonstrating that it holds true during the collision of algorithm.

Intialization:-

At the start of Each iteration of the outlet of loop the subarray before the current index 'i' is sorted, and all element in this subarray are smaller than or in Equal to any element in the unsolved portion of array.

```
for i=0 to n-1 {

        minimum =i

}
```

Maintnance:-

Before the excution of inner loop, all the elements of before i are sorted The inner loop identifies the minimum element in the array and swaps the values accordingly

```
for (j= i+1 to n) {
        if (arr [j] (arr [minimum]] {
            minimum = j
        }
    }
swap [arr [i], arr [minimum]]
```

The algorithm sorts the element at index P which belongs at the index minimum. The outer loop runs from 0 to n-1, iterating through array.

Termination:

Once the outer loop completes the algorithm terminates, the entire array is sorted, and here are no infinite loops of conditions that would prevent the completion.