Homework2 question1

Id -19720

Name-sravan kumar reddy chethireddy

What is the Big-O Time Complexity Analysis of BubbleSort? LC

- Process
  - Step 1: Please use Loop Analysis method to analyze the function

    void bubbleSort(int arr[])

    Please explain your answer.

```
class BubbleSort
{
    void bubbleSort(int arr[])
    {
        int n = arr.length;
        for (int i = 0; i < n-1; i++)
            for (int j = 0; j < n-i-1; j++)
                if (arr[j] > arr[j+1])
                {
                    // swap arr[j+1] and arr[i]
                    int temp = arr[j];
                    arr[j] = arr[j+1];
                    arr[j+1] = temp;
                }
    }

    /* Prints the array */
    void printArray(int arr[])
    {
        int n = arr.length;
        for (int i=0; i<n; ++i)
            System.out.print(arr[i] + " ");
        System.out.println();
    }

    // Driver method to test above
    public static void main(String args[])
    {
        BubbleSort ob = new BubbleSort();
        int arr[] = {64, 34, 25, 12, 22, 11, 90};
        ob.bubbleSort(arr);
        System.out.println("Sorted array");
        ob.printArray(arr);
    }
```

```
}
```

```
for (int i = 0; i < n-1; i++) =>O(n) * O(n)=O(n²)
            for (int j = 0; j < n-i-1; j++)=>O(n)
```

**the outer loop iterates from 0 to n-2**

**In the first iteration of the outer loop, the inner loop runs `n-1` times.**

**In the second iteration of the outer loop, the inner loop runs `n-2` times.**

**In the third iteration of the outer loop, the inner loop runs `n-3` times.**

**...**

**In the last iteration of the outer loop, the inner loop runs 1 time.**

```
for (int i=0; i<n; ++i)
```

**loop iterates from 0 to n-1 therefore BigO = O(n)**

BigO of the Bubble sort=$O(n^2)$ +$O(n)$=$O(n^2)$