Project 1

Name: Ankit R Mahale

"I have done this assignment completely on my own. I have not copied it, nor have I given my solution to anyone else. I understand that if I am involved in plagiarism or cheating I will have to sign an official form that I have cheated and that this form will be stored in my official university record. I also understand that I will receive a grade of 0 for the involved assignment for my first offense and that I will receive a grade of "F" for the course for any additional offense."

B# : B00594269

Sign: Ankit Mahale

Date:

2. Using the Method 1 or Method 2 discussed in class, compute the instruction counts for your insertion sort, counting sort, and merge sort implementations. Clearly state which method you use for instruction counts. (Note: You do not have to do this for quick sort.) Ensure that your counts are $\Theta(n^2)$ for insertion sort, $\Theta(n)$ for counting sort, and $\Theta(nlgn)$ for merge sort. If not, either your implementation is suboptimal or your instruction counts are incorrect. Correct any problems before submitting your assignment

Proof: $\Theta(n^2)$ for insertion sort

Barometer Operations:

```
for (i = 2 ; i <= n; i++)
{
        for(j=i-1;j > 0 && arr[j] < arr[j-1]; j--)
        {
                temp = arr[j];
                arr[j] = arr[j-1];
                arr[j-1] = temp;

        }
}
```

$$W(n) = \sum_{i=2}^{n} \sum_{j=0}^{i-1} 1 = \sum_{i=2}^{n} (i-1) = (n(n-1))/2 \in \Theta(n^2)$$

Proof: $\Theta(n)$ for counting sort

```
for(i=0;i<n;++i)
{
    count[arr[i]]=count[arr[i]]+1;
}
```

$$W(n) = \sum_{i=0}^{n} 1 = n \in \Theta(n)$$

Proof: $\Theta(nlgn)$ for merge sort

T(n)= Number of comparisons to merge sort an input array of size n

= T(n/2) + T(n/2) + n

T(n/2) = right half

T(n/2) = left half

n= merge

Divide eq(1) by N on both sides

T(n)/n = 2 T(n/2)/n + 1

= T(n/2) / (n/2) + 1

= T(n/4) / (n/4) + 1 + 1

= T(n/8) / (n/8) + 1 + 1 + 1

= T(n/16) / (n/16) + 1 + 1 + 1

= T(n/32) / (n/32) + 1 + 1 + 1

…….

= T(n/n)/(n/n) + 1 +1 … + 1

=lgn

T(n) = nlgn $\in$ Θ(nlgn)