AOA PRACTICAL LAB 1

Name: Brendan Lucas, Roll No:8953, Div: SE Comp B

Source Code:

```
#include <iostream>
#include <cstdlib>
using namespace std;

/int*/void insertionsort(int *arr,int n)
{
	//Declaring variables
	int j,i,num;

	//First for loop
	for(i=0;i<n;i++)
	{
		//Insert current element in sorted array
		num=arr[i];

		//assign j to current element index as index counter
		j=i;

		//shifting all elements greater than current element.
		while(j>0&&num<arr[j-1])
		{
			//Shift the element to right position
			arr[j]=arr[j-1];

			//decrease index counter by 1
			j--;
		}

		//place current element at appropriate position
		arr[j]=num;
	}
//	return arr;
}

int main(void)
{

	//Declering variables
	int n,*arr,i;

	//Asking for No. of elements in array
	cout<<"Enter the no of elements in array:-\n";
	cin>>n;

	//Creating array of given size using malloc
	arr=(int*)malloc(sizeof(int)*n);

	//Taking input of elements of array
	cout<<"Enter the array\n";
	for(i=0;i<n;i++)
	{
		cin>>arr[i];
	}

	//Sorting the array
	//bubblesort(arr, n);
	insertionsort(arr, n);
	//quicksort(arr,0, n-1);

	//Displaying the final result
	cout<<"\nThe Sorted array is:- \n";
	for(i=0;i<n;i++)
	{
		cout<<arr[i]<<" ";
	}
	return 0;
}
```

Q2 Insertion Sort :-

Code :-

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

Time Analysis :-

Best case :- Values are in ascending order

pass1 Insert value at position a[1] = 1 unit

pass2 Insert value at position a[2] = 1 unit

~~pass3~~

pass(n-1) Insert value at position a[n-1] = 1 unit

Total no of comparisons = n - 1

$f(n) = O(n-1)$

$= O(n)$.

Worst case :- Values are in descending order.

pass 1 No of comparisions = 1.

pass 2 No of comparsions = 2.

⋮

pass(n-1) No of comparsions = (n-1)

Total No of comparsions $= (n-1) + (n-2) + \ldots\ldots 2+1$

$= \frac{n(n \cancel{+} -1)}{2}$

$= 0.5n^2 - 0.5n$

$f(n) = O(0.5n^2 - 0.5n)$

$= O(n^2)$.

Average case :- Values are placed in halfway or in between.

$= \frac{1}{2} \frac{(n-1)(n)}{2} = \frac{n(n-1)}{4} = 0.25n^2 - 0.25n$

$\therefore f(n) = O(0.25n^2 - 0.25n)$

$= O(n^2)$.