ASSIGNMENT

MERGE SORT

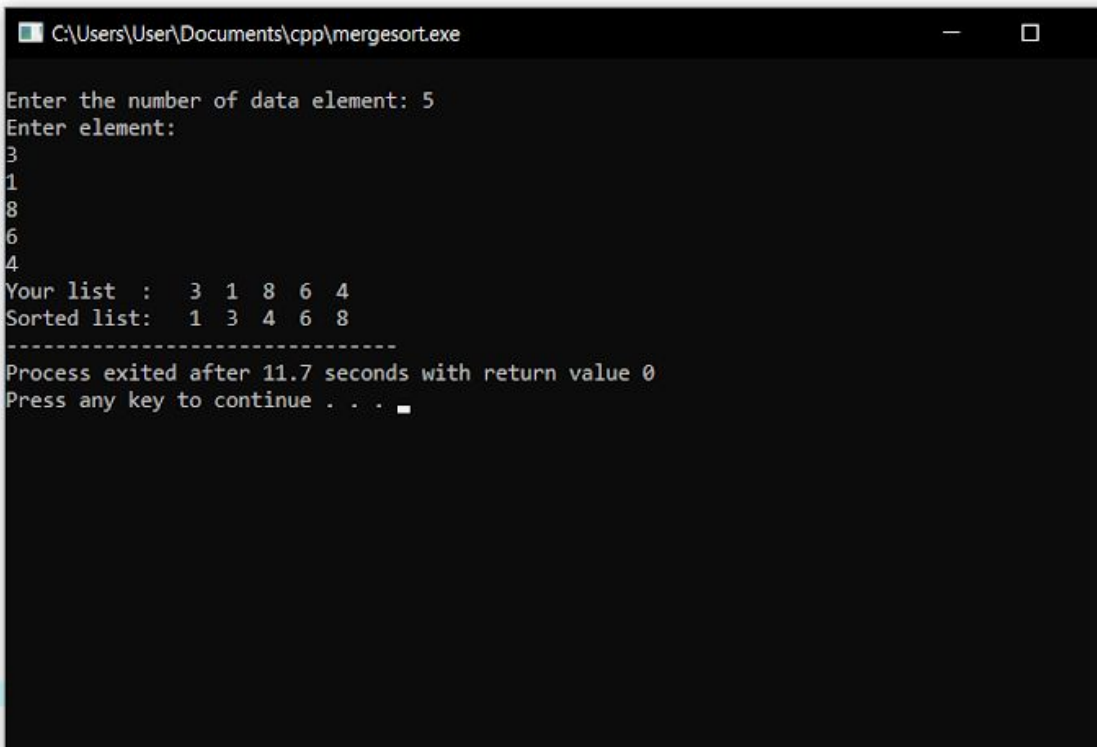
SOURCE CODE:

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

void Merge(int *a, int low, int high, int mid)
{
	int i, j, k, temp[high-low+1];
	i = low;
	k = 0;
	j = mid + 1;

	while (i <= mid && j <= high)
	{
		if (a[i] < a[j])
		{
			temp[k] = a[i];
			k++;
			i++;
		}
		else
		{
			temp[k] = a[j];
			k++;
			j++;
		}
	}

	while (i <= mid)
	{
		temp[k] = a[i];
		k++;
		i++;
	}

	while (j <= high)
	{
		temp[k] = a[j];
		k++;
		j++;
	}
```

```
        for (i = low; i <= high; i++)
        {
                a[i] = temp[i-low];
        }
}


void MergeSort(int *a, int low, int high)
{
        int mid;
        if (low < high)
        {
                mid=(low+high)/2;
                MergeSort(a, low, mid);
                MergeSort(a, mid+1, high);

                Merge(a, low, high, mid);
        }
}

int main()
{
        int n, i;
        cout<<"\nEnter the number of data element: ";
        cin>>n;

        int arr[n];
        cout<<"Enter element:\n";
   for(i = 0; i < n; i++)
        {
     cin>>arr[i];
   }

   cout<<"Your list  :   ";
   for(i = 0; i < n; i++)
   {
     cout<<arr[i]<<"  ";
   }

        MergeSort(arr, 0, n-1);

        cout<<"\nSorted list: ";
   for (i = 0; i < n; i++)
     cout<<"  "<<arr[i];

        return 0;
}
```

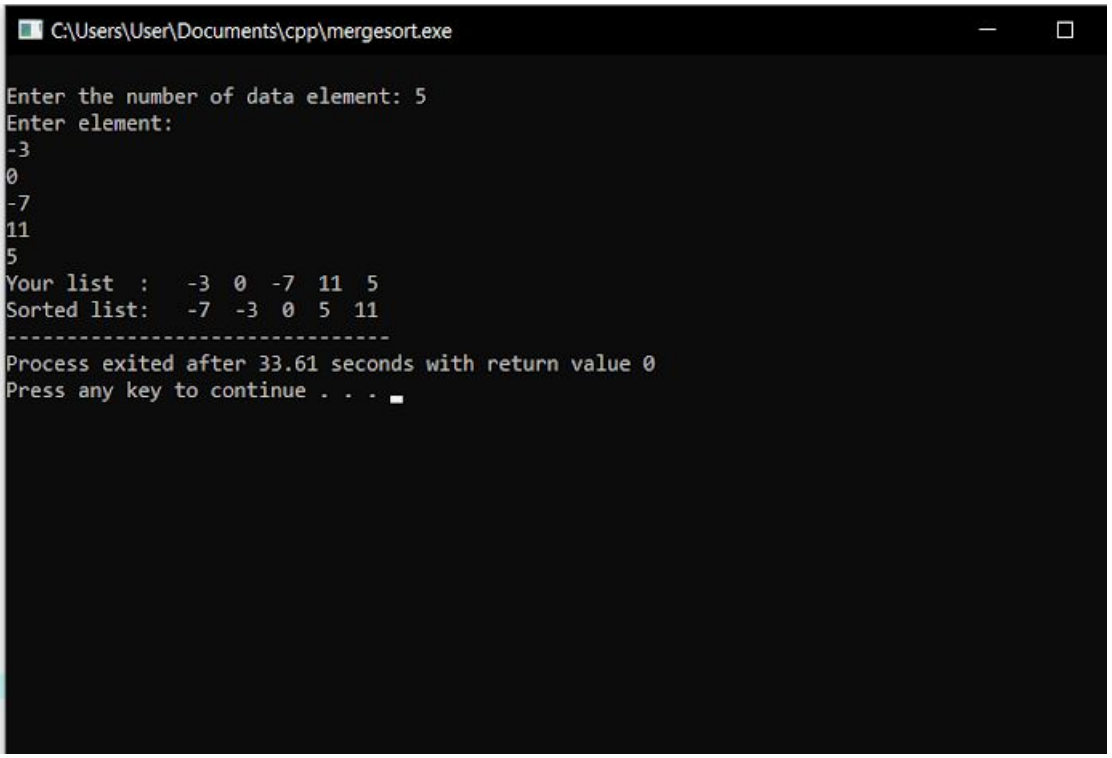
OUTPUT :

1.


C:\Users\User\Documents\cpp\mergesort.exe
Enter the number of data element: 5
Enter element:
3
1
8
6
4
Your list : 3 1 8 6 4
Sorted list: 1 3 4 6 8

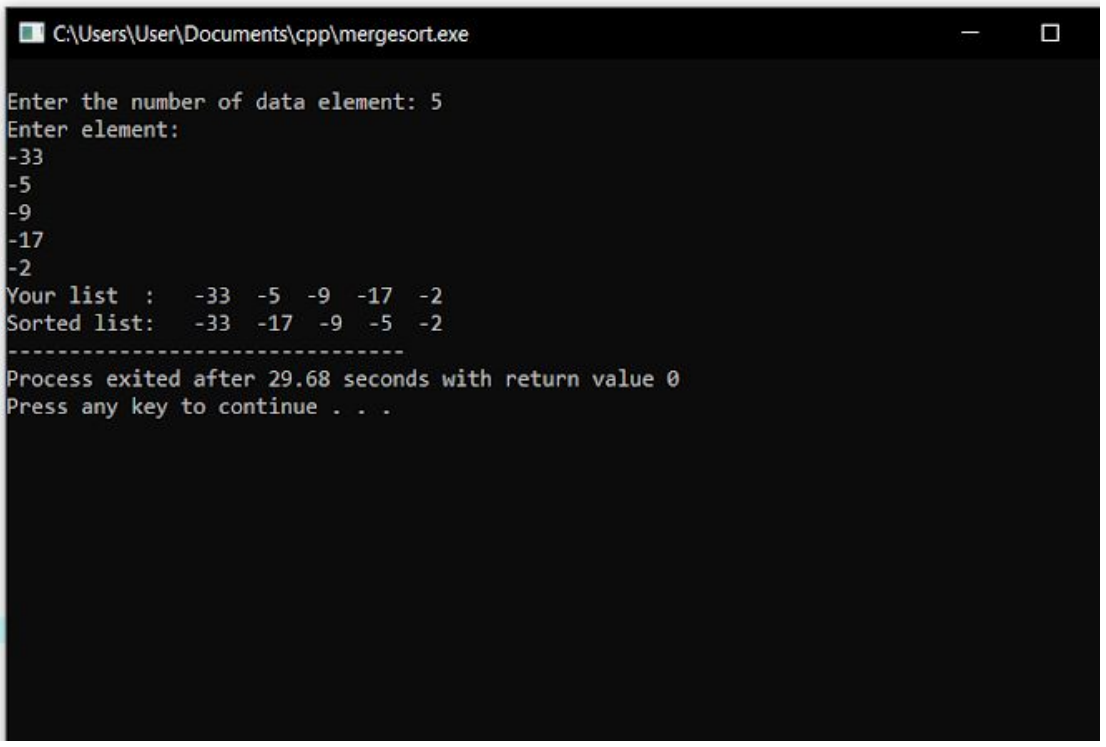
Process exited after 11.7 seconds with return value 0
Press any key to continue . . .

2.


C:\Users\User\Documents\cpp\mergesort.exe
Enter the number of data element: 5
Enter element:
-3
0
-7
11
5
Your list : -3 0 -7 11 5
Sorted list: -7 -3 0 5 11

Process exited after 33.61 seconds with return value 0
Press any key to continue . . .

3.



Time complexity : O(n*Log n)
in all the 3 cases (worst, average and best)

Space complexity : O(n)