```
/*WAP to implement Merge Sorting Algorithm*/
#include<iostream>
using namespace std;
void merge(int A[],int beg,int mid,int end)
{
   int i=beg;
   int j=mid+1;
   int index=beg;
   int temp[end+1],k;
   while(i<=mid && j<=end)
   {
      if(A[i]<A[j])
      {
         temp[index]=A[i];
         i++;
      }
      else
      {
         temp[index]=A[j];
         j++;
      }
      index++;
   }
   if(i>mid)
   {
      while(j<=end)
      {
         temp[index]=A[j];
         index++;
         j++;
      }
   }
   else
   {
      while(i<=mid)
      {
         temp[index]=A[i];
```

```
            index++;
            i++;
        }
    }
    k=beg;
    while(k<index)
    {
        A[k]=temp[k];
        k++;
    }
}
void merge_sort(int A[], int beg,int end)
{
    int mid;
    if(beg<end)
    {
        mid=(beg+end)/2;
        merge_sort(A,beg,mid);
        merge_sort(A,mid+1,end);
        merge(A,beg,mid,end);
    }
}
int main()
{
    int n;
    cout<<"enter number of items to be sorted:: ";
    cin>>n;
    int arr[n];
    cout<<"enter "<<n<<" items:: "<<endl;
    for(int i=0; i<n; i++)
        cin>>arr[i];
    merge_sort(arr,0,n-1);
    cout<<"the sorted items are:: "<<endl;
    for(int i=0; i<n; i++)
        cout<<arr[i]<<endl;
    return 0;
}
```