AM.SC.P2CSC19034

SRUTHY P R

**SORTING ELEMENTS SET 2**

MERGE SORT:

#include<iostream>

using namespace std;

void Merge(int A[] ,int start,int mid,int end)

{

int i,j,k;

int n1=mid-start+1;

int n2=end-mid;

int L[30] , R[30];

for(i=0;i<n1;i++)

{

L[i]=A[start +i];

}

for (j=0;j<n2;j++)

{

R[j]=A[mid+j+1];

}

i=0;j=0;

for(k=start;i<n1&& j<n2;k++)

{

if (L[i] <R[j])

{

A[k]=L[i++];

}

else

{

A[k]=R[j++];

}

}

while(i<n1)

{

A[k++]=L[i++];

}

while(j<n2)

{

A[k++]=R[j++];

}

}

void Merge\_sort(int A[],int start,int end)

{

int mid;

if(start <end)

{

mid=(start +end)/2;

Merge\_sort(A,start,mid);

Merge\_sort(A,mid+1,end);

Merge(A,start,mid,end);

}

}

int main()

{

int n,A[30];

cout<<"\n enter the size of array\n";

cin>>n;

cout<<"\n enter array elements\n";

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

{

cin>>A[i];

}

Merge\_sort(A,0,n-1);

cout<<"\n after sorting\n";

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

{

cout<<A[i]<<"";

cout<<endl;

}

}

Output:



QUICK SORT:

#include <iostream>

using namespace std;

void quick\_sort(int[],int,int);

int partition(int[],int,int);

int main()

{

int a[50],n,i;

cout<<"Number of elements:";

cin>>n;

cout<<"\nEnter the elements:";

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

cin>>a[i];

quick\_sort(a,0,n-1);

cout<<"\nArray after sorting:";

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

cout<<a[i]<<" ";

return 0;

}

void quick\_sort(int a[],int l,int u)

{

int j;

if(l<u)

{

j=partition(a,l,u);

quick\_sort(a,l,j-1);

quick\_sort(a,j+1,u);

}

}

int partition(int a[],int l,int u)

{

int v,i,j,temp;

v=a[l];

i=l;

j=u+1;

do

{

do

i++;

while(a[i]<v&&i<=u);

do

j--;

while(v<a[j]);

if(i<j)

{

temp=a[i];

a[i]=a[j];

a[j]=temp;

}

}while(i<j);

a[l]=a[j];

a[j]=v;

return(j);

}

Output:

