**ROLL NO:-45**

**NAME : Harshit Atul Chilvirwar**

**PRACTICAL NO:-**

**PRACTICAL NAME :- IMPLEMENTATION OF MERGE SORT**

#include "iostream.h"

#include "conio.h"

#include "stdlib.h"

class LIST

{

int \*A,size;

public:

LIST(int);

void SET\_LIST();

void VIEW\_LIST();

void MERGE\_SORT(int,int);

void MERGE(int,int,int);

};

LIST::LIST(int par)

{

size=par;

A =new int[size+1];

}

void LIST::SET\_LIST()

{

for(int i=1;i<=size;i++)

A[i]=random(50);

}

void LIST::VIEW\_LIST()

{

cout<<"List elements are : ";

for(int i=1;i<=size;i++)

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

}

void LIST::MERGE(int low,int mid,int high)

{

int \*B=new int[size+1];

int h=low,j=mid+1;

int i=low;

while(h<=mid && j<=high)

{

if(A[h]<A[j])

{

B[i]=A[h];

h=h+1;

i=i+1;

}

else

{

B[i]=A[j];

j=j+1;

i=i+1;

}

}

if(h>mid)

{

for(int k=j;k<=high;k++)

{

B[i]=A[k];

i=i+1;

}

}

else

{

for(int k=h;k<=mid;k++)

{

B[i]=A[k];

i=i+1;

}

}

for(int k=low; k<=high; k++)

{

A[k]=B[k];

}

delete B;

}

void LIST::MERGE\_SORT(int low,int high)

{

if(low<high)

{

int mid=(low+high)/2;

MERGE\_SORT(low,mid);

MERGE\_SORT(mid+1,high);

MERGE(low,mid,high);

}

}

void main()

{

int n;

clrscr();

cout<<"\n Enter size of array : ";

cin>>n;

LIST obj(n);

obj.SET\_LIST();

cout<<endl<<"List before sorting : \n";

obj.VIEW\_LIST();

obj.MERGE\_SORT(1,n);

cout<<endl<<"List after sorting : \n";

obj.VIEW\_LIST();

getch();

}