***BUBBLE SORT***

#include<bits/stdc++.h>

using namespace std;

int main()

{int v[100],i,j,n,x;

cin>>n;

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

cin>>v[i];

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

for(j=0;j<n-1-i;++j)

if(v[j]>v[j+1])

{

x=v[j];v[j]=v[j+1];v[j+1]=x;

}

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

cout<<v[i]<<' ';

}

Input:

8

34 56 12 89 901 78 4 366

Execution time: 6.254 s

***COUNT SORT***

#include<bits/stdc++.h>

using namespace std;

int main()

{int v[100],i,j,n,a[100],max=0,m=0;

cin>>n;

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

{

cin>>v[i];

if(max<v[i])

max=v[i];

}

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

a[i]=0;

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

a[v[i]]++;

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

{

for(j=1;j<=a[i];++j)

cout<<i<<' ';

}

}

Input:

10

1 2 3 2 3 4 5 8 1 3

Execution time: 13.267 s

***RADIX SORT***

#include<bits/stdc++.h>

using namespace std;

int main()

{int v[100],i,j,n,k=0,a[100],x=1,m=10,max=0,p=0;

cin>>n;

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

{

cin>>v[i];

if(max<v[i])

max=v[i];

}

while(max!=0)

{

p++;

max/=10;

}

while(x<=p){

for(i=0;i<=9;++i)

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

if((v[j]/(m/10))%10==i)

{

a[k]=v[j];

++k;

}

k=0;

++x;

m\*=10;

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

v[j]=a[j];

}

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

cout<<a[i]<<' ';

}

Input:

8

34 56 12 89 901 78 4 366

Execution time: 3.192 s

***MERGE SORT***

#include<bits/stdc++.h>

using namespace std;

int MS(int v[],int stg,int dr)

{int n,p,a1[100],a2[100],i,j,c=1;

if(stg>=dr)

return 0;

p=n/2;

MS(v,stg,p);

MS(v,p+1,dr);

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

a1[i]=v[i];

j=0;

for(i=p+1;i<n;++i)

{a2[j]=v[i];

j++;}

i=0;j=0;c=0;

while(i<=p&&j<n-p)

{

if(a1[i]<=a2[j])

{

v[c]=a1[i];

i++;

}

else

{

v[c]=a2[j];

j++;

}

c++;

}

while(i<p)

{

v[c]=a1[i];

++i;++c;

}

while(j<n-p)

{

v[c]=a2[j];

++j;++c;

}

}

int main()

{int n,i,j,v[100];

cin>>n;

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

cin>>v[i];

MS(v,0,n-1);

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

cout<<v[i]<<' ';

}