

```

1: #include<iostream>
2: #include<stdlib.h>
3: #include<omp.h>
4: using namespace std;
5:
6:
7: void mergesort(int a[],int i,int j);
8: void merge(int a[],int i1,int j1,int i2,int j2);
9:
10: void mergesort(int a[],int i,int j)
11: {
12:     int mid;
13:     if(i<j)
14:     {
15:         mid=(i+j)/2;
16:
17:         #pragma omp parallel sections
18:         {
19:
20:             #pragma omp section
21:             {
22:                 mergesort(a,i,mid);
23:             }
24:
25:             #pragma omp section
26:             {
27:                 mergesort(a,mid+1,j);
28:             }
29:         }
30:
31:         merge(a,i,mid,mid+1,j);
32:     }
33: }
34:
35:
36: void merge(int a[],int i1,int j1,int i2,int j2)
37: {
38:     int temp[1000];
39:     int i,j,k;
40:     i=i1;
41:     j=i2;
42:     k=0;
43:
44:     while(i<=j1 && j<=j2)
45:     {
46:         if(a[i]<a[j])
47:         {
48:             temp[k++]=a[i++];
49:         }
50:         else
51:         {
52:             temp[k++]=a[j++];
53:         }
54:     }
55:

```

```

56: while(i<=j1)
57: {
58:     temp[k++]=a[i++];
59: }
60:
61: while(j<=j2)
62: {
63:     temp[k++]=a[j++];
64: }
65:
66: for(i=i1,j=0;i<=j2;i++,j++)
67: {
68:     a[i]=temp[j];
69: }
70: }
71:
72:
73: int main()
74: {
75:     int *a,n,i;
76:     cout<<"\n enter total no of elements=>";
77:     cin>>n;
78:     a= new int[n];
79:
80:     cout<<"\n enter elements=>";
81:     for(i=0;i<n;i++)
82:     {
83:         cin>>a[i];
84:     }
85:
86:     mergesort(a, 0, n-1);
87:     cout<<"\n sorted array is=>";
88:     for(i=0;i<n;i++)
89:     {
90:         cout<<"\n"<<a[i];
91:     }
92:     return 0;
93: }

```