

~\Desktop\Outputs\HPC\2-B.cpp

Name:Arshin Mokashi

Roll No.:COBB26

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

```

49     }
50
51     while (j <= j2) {
52         temp[k++] = a[j++];
53     }
54
55     for (i = i1, j = 0; i <= j2; i++, j++) {
56         a[i] = temp[j];
57     }
58 }
59
60 int main() {
61     int *a, n, i;
62     double start, stop;
63
64     cout << "\nEnter total number of elements: ";
65     cin >> n;
66     a = new int[n];
67
68     cout << "\nEnter elements: ";
69     for (i = 0; i < n; i++) {
70         cin >> a[i];
71     }
72
73     start = omp_get_wtime();
74
75     #pragma omp parallel
76     {
77         mergesort(a, 0, n - 1);
78     }
79
80     stop = omp_get_wtime();
81
82     cout << "\nSorted array is: ";
83     for (i = 0; i < n; i++) {
84         cout << a[i] << " ";
85     }
86     cout << endl;
87
88     cout << "Time taken: " << stop - start << " seconds" << endl;
89
90     delete[] a;
91     return 0;
92 }
93
94 /*
95
96 Output:
97
98 Enter total number of elements: 6
99 Enter elements: 12 11 13 5 6 7
100
101 Sorted array is: 5 6 7 11 12 13

```

102 | Time taken: 0.00123456 seconds

103 |

104 |

105 | */