File: oddEvenMergeSort.cpp

```
#include<omp.h>
#include<iostream>
using namespace std;
int n, a[100];

void display() {
     for(int i = 0; i < n; i++) {
          cout<<a[i]<<"\t";
     }
}

void oddEvenMerge(int lo, int hi, int r) {
     int step = r * 2;
     if(step < hi) {
          #pragma omp parallel sections
          {
               #pragma omp section
               {
                    oddEvenMerge(lo, hi, step);
               }
               #pragma omp section
               {
                    oddEvenMerge(lo+r, hi, step);
               }
          }
          for(int i = lo + r; i < lo + hi - r; i = i + step) {
               if(a[i] > a[i+r]) {
                    int t = a[i];
                    a[i] = a[i+r];
                    a[i+r] = t;
               }
          }
     }
     else {
          if(a[lo] > a[lo+r]) {
               int t = a[lo];
               a[lo] = a[lo+r];
               a[lo+r] = t;
          }
     }
}

void oddEvenMergeSort(int lo, int n) {
     if(n > 1) {
          int mid = n / 2;
          oddEvenMergeSort(lo, mid);
          oddEvenMergeSort(lo + mid, mid);
          oddEvenMerge(lo, n, 1);
     }
}
```

```
int main() {
     cout<<"\nEnter Number of Elements to be Sorted (power of 2): ";
     cin>>n;
     cout<<"\nEnter Elements: ";
     for(int i = 0; i < n; i++) {
          cin>>a[i];
     }
     oddEvenMergeSort(0,n);
     cout<<"\nSorted elements are: ";
     display();
     cout<<"\n\n";
}
```

#OUTPUT: