Question02.cpp

```
1 // <--- DSA LAB 8 --->
 2
   // <--- 02 --->
 3
    // Write a program to implement a recursive version of quicksort. Run it for some sample
 4
 5
   // data.
 6
 7
    #include<iostream>
    using namespace std;
8
9
10
    int partition( int arr[], int s, int e) {
11
12
13
        int pivot = arr[s];
14
15
        int cnt = 0;
16
        for(int i = s+1; i<=e; i++) {</pre>
            if(arr[i] <=pivot) {</pre>
17
18
                cnt++;
19
20
        }
21
22
        //place pivot at right position
23
        int pivotIndex = s + cnt;
24
        swap(arr[pivotIndex], arr[s]);
25
26
        //left and right wala part smbhal lete h
27
        int i = s, j = e;
28
        while(i < pivotIndex && j > pivotIndex) {
29
30
            while(arr[i] <= pivot)</pre>
31
32
33
                i++;
34
35
36
            while(arr[j] > pivot) {
37
                j--;
38
            }
39
40
            if(i < pivotIndex && j > pivotIndex) {
41
                swap(arr[i++], arr[j--]);
42
            }
43
44
        }
45
46
        return pivotIndex;
47
48
    }
49
50
    void quickSort(int arr[], int s, int e) {
51
52
        //base case
53
        if(s >= e)
```

```
54
            return ;
55
56
        //partitioon karenge
57
        int p = partition(arr, s, e);
58
59
        //left part sort karo
60
        quickSort(arr, s, p-1);
61
62
        //right wala part sort karo
63
        quickSort(arr, p+1, e);
64
65
    }
66
67
    int main() {
68
        int arr[10] = {2,4,1,6,1000,8,5,0,8,10};
69
70
        int n = 10;
71
        quickSort(arr, 0, n-1);
72
73
        for(int i=0; i<n; i++)</pre>
74
75
76
            cout << arr[i] << " ";</pre>
        } cout << endl;</pre>
77
78
79
80
        return 0;
81 }
```