



```
1 // Objective: - Program for Quick Sort.
2
3 #include <stdio.h>
4 void swap(int *a, int *b)
5 {
6     int t = *a;
7     *a = *b;
8     *b = t;
9 }
10 int partition(int arr[], int low, int high)
11 {
12     int pivot = arr[high];
13     int i = (low - 1);
14     for (int j = low; j <= high - 1; j++)
15     {
16         if (arr[j] < pivot)
17         {
18             i++;
19             swap(&arr[i], &arr[j]);
20         }
21     }
22     swap(&arr[i + 1], &arr[high]);
23     return (i + 1);
24 }
25 void quickSort(int arr[], int low, int high)
26 {
27     if (low < high)
28     {
29         int pi = partition(arr, low, high);
30         quickSort(arr, low, pi - 1);
31         quickSort(arr, pi + 1, high);
32     }
33 }
34 int main()
35 {
36     int arr[] = {30, 52, 5, 13, 25, 44};
37     int N = sizeof(arr) / sizeof(arr[0]);
38
39     quickSort(arr, 0, N - 1);
40
41     printf("Sorted array: \n");
42     for (int i = 0; i < N; i++)
43         printf("%d ", arr[i]);
44     return 0;
45 }
46
47
48 // Output :
49 // Sorted array:
50 // 5 13 25 30 44 52
```