S.No: 7	Exp. Name: ***Quick Sort***	**Date:2025-09-16**

Aim:

Write a C program that implements the Quick Sort to sort a given list of integers in ascending order.

Input Format

- The program should prompt the user to enter the size of the array and the elements of the array.

Output Format

- After sorting the array, the program should display the sorted array.

Note: Partial code is already given to you in the editor. Fill in the remaining code to achieve the task.

Source Code:

quickSort.c

```
#include <stdio.h>

void display(int arr[15], int n) {
   int i;
   for (i = 0; i < n; i++)
      printf("%d ", arr[i]);
   printf("\n");
}
   void swap(int* a, int* b){
      int temp = *a;
      *a=*b;
      *b=temp;

}
int partition(int arr[15], int lb, int ub) {
   int p = arr[ub];
   int i = (lb-1);
   for(int j=lb; j<ub; j++){
      if(arr[j]<=p){
         i++;
         swap(&arr[i],&arr[j]);
      }
   }
   swap(&arr[i+1], &arr[ub]);
   return(i+1);



   //Type your content here






}
void quickSort(int arr[15], int low, int high) {
   int j;
   if (                low < high ) {
      j = partition(        arr,low,high   );
      quickSort(           arr,low,j-1    );
      quickSort(            arr,j+1,high   );
   }
}

void main() {
   int arr[15], i, n;
   printf("Enter array size : ");
   scanf("%d", &n);
   printf("Enter %d elements : ", n);
   for (i = 0; i < n; i++) {
      scanf("%d", &arr[i]);
   }
   printf("Before sorting the elements are : ");
   display(arr, n);
   quickSort(arr, 0, n - 1);
   printf("After sorting the elements are : ");
   display(arr, n);
}
```

Execution Results - All test cases have succeeded!

Test Case - 1

User Output

```
Enter array size :  6
Enter 6 elements :  6 2 8 10 36 14
Before sorting the elements are : 6 2 8 10 36 14
After sorting the elements are : 2 6 8 10 14 36
```

Test Case - 2

User Output

```
Enter array size :  8
Enter 8 elements :  95 14 10 23 36 2 5 35
Before sorting the elements are : 95 14 10 23 36 2 5 35
After sorting the elements are : 2 5 10 14 23 35 36 95
```