```c
#include <stdio.h>

void printArray(int *A, int n)
{
    for (int i = 0; i < n; i++)
    {
        printf("%d ", A[i]);
    }
    printf("\n");
}

int partition(int A[], int low, int high)
{
    int pivot = A[low];
    int i=low+1;
    int j=high;
    int temp;

    do
    {

          for(;i<=high && A[i]<pivot;i++){

        }




          for(;j>=low && A[j]>pivot;j--){

        }

        if (i < j)
        {
            temp = A[i];
            A[i] = A[j];
            A[j] = temp;
        }
    } while (i < j);

    // Swap A[low] and A[j]
    temp = A[low];
    A[low] = A[j];
    A[j] = temp;
    return j;
}

void quickSort(int A[], int low, int high)
{
    int partitionIndex; // Index of pivot after partition

    if (low < high)
    {
        partitionIndex = partition(A, low, high);
        quickSort(A, low, partitionIndex - 1);  // sort left subarr
        quickSort(A, partitionIndex + 1, high); // sort right subar
    }
}

int main()
{
    //int A[] = {3, 5, 2, 13, 12, 3, 2, 13, 45};
    int A[] = {9, 4, 4, 8, 7, 5, 6};
    // 3, 5, 2, 13, 12, 3, 2, 13, 45
    // 3, 2, 2, 13i, 12, 3j, 5, 13, 45
    // 3, 2, 2, 3j, 12i, 13, 5, 13, 45 --> first call to partiti
    int n = 9;
    n =7;
    printArray(A, n);
    quickSort(A, 0, n - 1);
    printArray(A, n);
    return 0;
}
```