Code:

//Quick sort program

#include <stdio.h>

void swap(int\* a, int\* b) {

int temp = \*a;

\*a = \*b;

\*b = temp;

}

int partition(int arr[], int low, int high) {

int pivot = arr[high];

int I = (low – 1);

for (int j = low; j <= high – 1; j++) {

if (arr[j] < pivot) {

i++;

swap(&arr[i], &arr[j]);

}

}

swap(&arr[I + 1], &arr[high]);

return (I + 1);

}

void quickSort(int arr[], int low, int high) {

if (low < high) {

int pi = partition(arr, low, high);

// Print the current state of the array

printf(“Step: “);

for (int I = low; I <= high; i++) {

printf(“%d “, arr[i]);

}

printf(“\n”);

quickSort(arr, low, pi – 1);

quickSort(arr, pi + 1, high);

}

}

int main() {

int arr[] = {12, 11, 13, 5, 6, 7};

int n = sizeof(arr) / sizeof(arr[0]);

printf(“Original array: “);

for (int I = 0; I < n; i++) {

printf(“%d “, arr[i]);

}

printf(“\n\n”);

quickSort(arr, 0, n – 1);

printf(“\nSorted array: “);

for (int I = 0; I < n; i++) {

printf(“%d “, arr[i]);

}

printf(“\n”);

return 0;

}

/\* OUTPUT :

Original array: 12 11 13 5 6 7

Step: 5 6 7 12 11 13

Step: 5 6

Step: 12 11 13

Step: 11 12

Sorted array: 5 6 7 11 12 13

\*/