#include <stdio.h>

// Function to swap two elements

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

int temp = \*a;

\*a = \*b;

\*b = temp;

}

// Function to partition the array and return the pivot index

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);

}

// Function to perform quicksort

void quick\_sort(int arr[], int low, int high) {

if (low < high) {

// Partitioning index

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

// Separately sort elements before partition and after partition

quick\_sort(arr, low, pi - 1);

quick\_sort(arr, pi + 1, high);

}

}

// Function to print the array

void print\_array(int arr[], int n) {

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

printf("%d ", arr[i]);

printf("\n");

}

int main() {

int arr[] = {10, 7, 8, 9, 1, 5};

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

printf("Original array: ");

print\_array(arr, n);

quick\_sort(arr, 0, n - 1);

printf("Sorted array: ");

print\_array(arr, n);

return 0;

}