// Java program for 3-way quick sort

class QuickThreeWay {

static int i, j;

/\* This function partitions array in three parts

all elements smaller than pivot

all elements equal to pivot

all elements greater than pivot \*/

static void partition(int a[], int l, int r) {

i = l - 1;

j = r;

int p = l - 1, q = r;

int v = a[r];

while (true) {

while (a[++i] < v)

;

while (v < a[--j])

if (j == l)

break;

if (i >= j)

break;

int temp = a[i];

a[i] = a[j];

a[j] = temp;

if (a[i] == v) {

p++;

temp = a[i];

a[i] = a[p];

a[p] = temp;

}

if (a[j] == v) {

q--;

temp = a[q];

a[q] = a[j];

a[j] = temp;

}

}

// Fixing Pivot

int temp = a[i];

a[i] = a[r];

a[r] = temp;

j = i - 1;

for (int k = l; k < p; k++, j--) {

temp = a[k];

a[k] = a[j];

a[j] = temp;

}

i = i + 1;

for (int k = r - 1; k > q; k--, i++) {

temp = a[i];

a[i] = a[k];

a[k] = temp;

}

}

// 3-way partition based quick sort

static void quicksort(int a[], int l, int r) {

if (r <= l)

return;

i = 0;

j = 0;

partition(a, l, r);

quicksort(a, l, j);

quicksort(a, i, r);

}

static void printarr(int a[], int n) {

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

System.out.printf("%d ", a[i]);

System.out.printf("\n");

}

public static void main(String[] args) {

int a[] = {4, 9, 4, 4, 1, 9, 4, 4, 9, 4, 4, 1, 4};

int size = a.length;

// Function Call

printarr(a, size);

quicksort(a, 0, size - 1);

printarr(a, size);

}

}