DATA STRUCTURES AND ALGORITHMS

LAB ASSIGNMENT-3

NAME – KAPAROTU VENKATA SURYA THARANI

COURSE – AIDE

SECTION - “A”

USN ID – 22BTRAD018

Write two different (recursive and non-recursive) functions to implement quick sort.

# Recursive function to implement quick sort: Code:

**package** p1;

**import** java.util.Arrays;

**import** java.util.Scanner;

**public class** QuickSort {

**int** partition(**int** ar1[], **int** low, **int** high)

{

**int** pivot= ar1[high];

**int** a= (low-1);

**for** (**int** i=low; i<high; i++)

{

**if** (ar1[i] < pivot)

{

a++;

**int** temp=ar1[a]; ar1[a]=ar1[i]; ar1[i]= temp;

}

}

**int** temp=ar1[a+1]; ar1[a+1]=ar1[high]; ar1[high]= temp; **return** a+1;

}

**void** quicksort(**int** ar1[], **int** low, **int** high)

{

**if** (low<high)

{

**int** part= partition(ar1, low, high); quicksort(ar1, low, part-1); quicksort(ar1, part+1, high);

}

}

**public static void** main(String[] args) {

Scanner sc=**new** Scanner(System.***in***);

**int** ar1[], i;

System.***out***.println("Enter array size:");

**int** size= sc.nextInt(); ar1=**new int**[size];

System.***out***.println("Enter the elements of the array: ");

**for**(i=0;i<size;i++)

{

ar1[i]=sc.nextInt();

}

QuickSort arr= **new** QuickSort(); arr.quicksort(ar1, 0, size-1); System.***out***.println("Sorted array: "); **for**(i=0;i<size;i++) System.***out***.print(ar1[i]+" ");

}

}

**Output:**



**Non-recursive function to implement quick sort: Code:**

**package** p1;

**public class** Quick\_Sort {

**void** swap\_val(**int** ar1[], **int** i, **int** j){

**int** temp = ar1[i]; ar1[i] = ar1[j]; ar1[j] = temp;

}

**int** partition(**int** ar1[], **int** low, **int** high){

**int** a = ar1[high];

**int** i = (low - 1);

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

**if** (ar1[j] <= a){ i++;

swap\_val(ar1, i, j);

}

}

swap\_val(ar1, i + 1, high);

**return** (i + 1);

}

**void** quick\_sort(**int** ar1[], **int** low, **int** high){

**int** l1[] = **new int**[high - low + 1];

**int** top = -1; l1[++top] = low; l1[++top] = high; **while** (top >= 0){

high = l1[top--];

low = l1[top--];

**int** p = partition(ar1, low, high);

**if** (p - 1 > low){ l1[++top] = low; l1[++top] = p - 1;

}

**if** (p + 1 < high){ l1[++top] = p + 1; l1[++top] = high;

}

}

}

**public static void** main(String args[]){ Quick\_Sort arr = **new** Quick\_Sort();

**int** arr1[] = {16,9,5,23,7,8,11,30,25,3};

**int** j;

System.***out***.println("Array before sorting: "); **for** (j = 0; j < arr1.length; ++j) System.***out***.print(arr1[j]+" ");

System.***out***.println(" "); arr.quick\_sort(arr1, 0, arr1.length - 1); **int** i;

System.***out***.println("Array after sorting: "); **for** (i = 0; i < arr1.length; ++i) System.***out***.print(arr1[i] + " ");

}

}

**Output:**

