

Question3.java

```
1 import java.io.*;
2 import java.util.*;
3 import java.lang.Math.*;
4
5 /**
6  * class ReadFile reads the input file and stores the values in an int
   array
7  */
8 class ReadFile {
9
10     private Scanner x;
11
12     /*
13      * openFile - checks and handles the exception of whether the file is
   found or
14      * not.
15      */
16     public void openFile() {
17         try {
18             x = new Scanner(new File("testQ3.txt"));
19         } catch (Exception e) {
20             System.out.println("File not found");
21         }
22     }
23
24     /*
25      * readFile() funtion reads the value from file and returns the values
   as an
26      * array of integers
27      */
28     public int[] readFile() {
29         int[] arr = new int[10];
30         int i = 0;
31         while (x.hasNext()) {
32             arr[i] = x.nextInt();
33             i++;
34         }
35         return arr;
36     }
37
38     public void closeFile() {
39         x.close();
40     }
41 }
```

Question3.java

```
40     }
41 }
42
43 /**
44  * Quick sort class via multiple methods, sorts the input values
45  */
46 class QuickSort {
47
48     /*
49      * arr[] - array of integers from input file; left - first half of the
      array
50      * right - second half of the array
51      */
52     int partition(int arr[], int left, int right) {
53         int i = left, j = right;
54         int pivot = arr[(left + right) / 2];
55         int tmp;
56
57         while (i <= j) {
58             while (arr[i] < pivot)
59                 i++;
60             while (arr[j] > pivot)
61                 j--;
62             if (i <= j) {
63                 tmp = arr[i];
64                 arr[i] = arr[j];
65                 arr[j] = tmp;
66                 i++;
67                 j--;
68             }
69         }
70         return i;
71     }
72
73     /*
74      * arr[] - array of integers from input file; left - first half of the
      array
75      * right - second half of the array
76      */
77     void sort(int arr[], int left, int right) {
78         int index = partition(arr, left, right);
79         if (left < index - 1)
```

Question3.java

```
80         sort(arr, left, index - 1);
81     if (index < right)
82         sort(arr, index, right);
83 }
84
85 /*
86  * arr[] - array of integers from input file;
87  */
88 void showList(int arr[]) {
89     for (int i = 0; i < arr.length; i++) {
90         System.out.print(arr[i] + " ");
91     }
92     System.out.println("\n");
93 }
94 }
95
96 public class Question3 {
97     public static void main(String[] args) {
98         ReadFile read = new ReadFile();
99
100         QuickSort qS = new QuickSort();
101         // Open file
102         read.openFile();
103         // Read file
104         int[] arr = read.readFile();
105
106         qS.showList(arr);
107         qS.partition(arr, 0, arr.length - 1);
108         qS.sort(arr, 0, arr.length - 1);
109         qS.showList(arr);
110
111         read.closeFile();
112     }
113 }
114
```