

```

1 package aufgabenblatt3;
2
3 import java.util.Arrays;
4
5 public class QuicksortA
6 {
7     private int[] a;
8     private int n;
9
10    public int vergleiche;
11    public int tausche;
12    public int zuweisungen;
13
14    public void sort(int[] a)
15    {
16        this.a=a;
17        n=a.length;
18        quicksort(0, n-1);
19    }
20
21    private void quicksort (int lo, int hi)
22    {
23        int i=lo, j=hi;
24        zuweisungen+=2;
25
26        // Vergleichselement x (pivotelement) bestimmen
27        int x=a[(lo+hi-lo)/2];
28
29        zuweisungen++;
30
31        // Aufteilung
32        while (i<=j)
33        {
34            vergleiche++;
35
36            while (a[i]<x) {
37                i++;
38                zuweisungen++;
39                vergleiche++;
40            }
41
42            while (a[j]>x) {
43                j--;
44                zuweisungen++;
45                vergleiche++;
46            }
47
48            if (i<=j)
49            {
50                exchange(i, j);
51                i++; j--;
52                zuweisungen+=2;
53                vergleiche++;
54            }
55        }
56    }
57
58    // Rekursion
59    if (lo<j) quicksort(lo, j);
60    if (i<hi) quicksort(i, hi);
61 }
62
63 private void exchange(int i, int j)
64 {
65     int t=a[i];
66     a[i]=a[j];
67     a[j]=t;
68     zuweisungen+=3;
69     tausche++;
70 }
71
72 public static void main(String[] args) {
73
74     //kleiner Test
75     int[] folge10 = {10,8,5,4,3,6,7,2,9,1};
76
77     int[] folge100 = new int[100];
78     for(int i=0;i<100;i++){
79         folge100[i]=i+1;
80     }
81
82     QuicksortA sorter = new QuicksortA();
83     sorter.sort(folge100);
84
85     System.out.println(folge10);
86 }
87
88 }

```