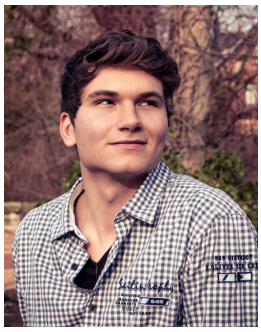
1 - 10

Übung 4



Martin Weber
1414353

Mike Hukiewitz
1423397

Aufgabe 1

Quellcode siehe MergeSort.java

Beispiel Protokoll von printruns():

```
Tape:[ 17 31 ] [ 5 59 ] [ 13 41 43 67 ] [ 11 23 29 47 ] [ 3 7 71 ] [ 2 19 57 ] [ 37 61 ]
TempTape1:[ 17 31 ] [ 13 41 43 67 ] [ 3 7 71 ] [ 37 61 ]
TempTape2:[ 5 59 ] [ 11 23 29 47 ] [ 2 19 57 ]
Tape:[ 5 17 31 59 ] [ 11 13 23 29 41 43 47 67 ] [ 2 3 7 19 57 71 ] [ 37 61 ]
TempTape1:[ 5 17 31 59 ] [ 2 3 7 19 57 71 ]
TempTape2:[ 11 13 23 29 41 43 47 67 ] [ 37 61 ]
Tape:[ 5 11 13 17 23 29 31 41 43 47 59 67 ] [ 2 3 7 19 37 57 61 71 ]
TempTape1:[ 5 11 13 17 23 29 31 41 43 47 59 67 ]
TempTape2:[ 2 3 7 19 37 57 61 71 ]
Tape:[ 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 57 59 61 67 71 ]
TempTape1:[ 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 57 59 61 67 71 ]
TempTape2:[ ]
Sorted Tape:[ 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 57 59 61 67 71 ]
```

Aufgabe 2

Quellcode siehe QuickSort3.java

	Arraylänge: **1024**	Arraylänge: **2048**	Arraylänge: **4096**
Anzahl **Vergleiche**	11225	25608	56897
Anzahl **Vertauschungen**	6610	14763	32391
Anzahl **Durchläufe**	1387	2773	5545

(Durchschnittliche Werte aus 100 Ausführungen)

```
51 23 1 0 23 7 8 5 14 17 8 3 7 7 2
1 0 [2] 23 23 7 8 5 14 17 8 3 7 7 51
[0] 1 2 23 23 7 8 5 14 17 8 3 7 7 51
0 1 2 23 23 7 8 5 14 17 8 3 7 7 [51]
0 1 2 7 5 3 7 [7] 14 17 8 23 8 23 51
0 1 2 7 5 3 [7] 7 14 17 8 23 8 23 51
0 1 2 [3] 5 7 7 7 14 17 8 23 8 23 51
0 1 2 3 5 [7] 7 7 14 17 8 23 8 23 51
0 1 2 3 5 7 7 7 14 17 8 23 8 [23] 51
0 1 2 3 5 7 7 7 8 [8] 14 23 17 23 51
0 1 2 3 5 7 7 7 8 8 14 [17] 23 23 51
```

Das Pivot-Element ist eingeklammert.

Aufgabe 3
Quellcode siehe ShellSort.java

	Arraylänge: **1024**	Arraylänge: **2048**	Arraylänge: **4096**
Anzahl **Vergleiche**	32985	126184	491612
Anzahl **Vertauschungen**	41136	142527	524339
Anzahl **Durchläufe**	4075	8171	16363

(Durchschnittliche Werte aus 100 Ausführungen)

```
Distanz 9: [8] 6 3 8 5 12 61 23 62 [9] 0 1 2 6 4 6 1
Distanz 9: 8 [0] 3 8 5 12 61 23 62 9 [6] 1 2 6 4 6 1
Distanz 9: 8 0 [1] 8 5 12 61 23 62 9 6 [3] 2 6 4 6 1
Distanz 9: 8 0 1 [2] 5 12 61 23 62 9 6 3 [8] 6 4 6 1
Distanz 9: 8 0 1 2 [5] 12 61 23 62 9 6 3 8 [6] 4 6 1
Distanz 9: 8 0 1 2 5 [4] 61 23 62 9 6 3 8 6 [12] 6 1
Distanz 9: 8 0 1 2 5 4 [6] 23 62 9 6 3 8 6 12 [61] 1
Distanz 9: 8 0 1 2 5 4 6 [1] 62 9 6 3 8 6 12 61 [23]
Distanz 7: [1] 0 1 2 5 4 6 [8] 62 9 6 3 8 6 [12] 61 23
Distanz 7: 1 [0] 1 2 5 4 6 8 [62] 9 6 3 8 6 12 [61] 23
Distanz 7: 1 0 [1] 2 5 4 6 8 62 [9] 6 3 8 6 12 61 [23]
Distanz 7: 1 0 1 [2] 5 4 6 8 62 9 [6] 3 8 6 12 61 23
Distanz 7: 1 0 1 2 [3] 4 6 8 62 9 6 [5] 8 6 12 61 23
Distanz 7: 1 0 1 2 3 [4] 6 8 62 9 6 5 [8] 6 12 61 23
Distanz 7: 1 0 1 2 3 4 [6] 8 62 9 6 5 8 [6] 12 61 23
Distanz 7: [1] 0 1 2 3 4 6 [8] 62 9 6 5 8 6 [12] 61 23
Distanz 7: 1 [0] 1 2 3 4 6 8 [61] 9 6 5 8 6 12 [62] 23
Distanz 7: 1 0 [1] 2 3 4 6 8 61 [9] 6 5 8 6 12 62 [23]
Distanz 4: [1] 0 1 2 [3] 4 6 8 [61] 9 6 5 [8] 6 12 62 [23]
Distanz 4: 1 [0] 1 2 3 [4] 6 8 61 [9] 6 5 8 [6] 12 62 23
Distanz 4: 1 0 [1] 2 3 4 [6] 8 61 9 [6] 5 8 6 [12] 62 23
Distanz 4: 1 0 1 [2] 3 4 6 [8] 61 9 6 [5] 8 6 12 [62] 23
Distanz 4: [1] 0 1 2 [3] 4 6 8 [61] 9 6 5 [8] 6 12 62 [23]
Distanz 4: 1 [0] 1 2 3 [4] 6 8 61 [9] 6 5 8 [6] 12 62 23
Distanz 4: 1 0 [1] 2 3 4 [6] 8 61 9 [6] 5 8 6 [12] 62 23
Distanz 4: 1 0 1 [2] 3 4 6 [5] 61 9 6 [8] 8 6 12 [62] 23
Distanz 4: [1] 0 1 2 [3] 4 6 5 [8] 9 6 8 [61] 6 12 62 [23]
Distanz 4: 1 [0] 1 2 3 [4] 6 5 8 [6] 6 8 61 [9] 12 62 23
Distanz 4: 1 0 [1] 2 3 4 [6] 5 8 6 [6] 8 61 9 [12] 62 23
Distanz 4: 1 0 1 [2] 3 4 6 [5] 8 6 6 [8] 61 9 12 [62] 23
Distanz 4: [1] 0 1 2 [3] 4 6 5 [8] 6 6 8 [23] 9 12 62 [61]
Distanz 1: [0] [1] [1] [2] [3] [4] [5] [6] [6] [6] [8] [8] [9] [12] [23] [61] [62]
```

Nur die eingeklammerten Werte werden abgeglichen und sortiert.