

Aufgabe 1

Bestimmen Sie die Laufzeit des folgenden Algorithmus:

```
for (i=0; i<n-1; i++) {A[i]=n-i;}      O(n)
for (j=0; j<n2; j++) {B[j]=0;}      O(n2)
```

zusammen ergibt sich $T(n) = O(n^2) + O(n) = O(n^2)$

Aufgabe 2

Bestimmen Sie die Laufzeit des folgenden Algorithmus:

```
for (i=0; i<n-1; i++) {
  for (j=0; j<n2; j++) {
    for (k=0; k<n; k++) {
      A[i][j]=B[k];
    }
  }
}
```

$T(n) = O(n^4)$

Aufgabe 3

Bestimmen Sie die Laufzeit des folgenden Algorithmus:

```
for (i=1; i<n-1;) {
  A[i]=i
  i=i*2;
}
```

$T(n) = O(\log n)$

Aufgabe 4

Gegeben ist ein sortiertes Feld mit dem Inhalt

ABCDHJLÖPQTUXZ.

Führen Sie die binäre Suche für C, Ö und Z aus. Geben Sie alle zum Verständnis wichtigen Zwischenschritte an!

Wir suchen C und beginnen in

$F[7]=L$, C ist kleiner, weitersuchen in $F[0..6]$

$F[3]=C$, C gefunden!

Wir suchen Ö und beginnen in

$F[7]=L$, Ö ist größer, weitersuchen in $F[8..14]$

$F[11]=T$, Ö ist kleiner, weitersuchen in $F[8..10]$

$F[9]=P$, Ö ist kleiner, weitersuchen in $F[8]$

$F[8]=Ö$, Ö gefunden!

Wir suchen Z und beginnen in

$F[7]=L$, Z ist größer, weitersuchen in $F[8..14]$

$F[11]=T$, Z ist größer, weitersuchen in $F[12..14]$

$F[13]=X$, Z ist größer, weitersuchen in $F[14]$

$F[14]=Z$, Z gefunden!

Aufgabe 5

Sortieren Sie mit Selection Sort die Folge HANNOVER. Geben Sie alle zum Verständnis wichtigen Zwischenschritte an!

1. H A N N O V E R
2. A H N N O V E R
3. A E N N O V H R
4. A E H N O V N R
5. A E H N O V N R
6. A E H N N V O R
7. A E H N N O V R
8. A E H N N O R V

Schritt 1: Wir haben die vollständige, unsortierte Zeichenfolge "HANNOVER". Der kleinste Buchstabe ist das "A".

Schritt 2: Das "A" wird mit dem Zeichen an der ersten Stelle getauscht. Somit wandert das "A" an Stelle 1, das "H" von Stelle 1 an Stelle 2. Das nächste Zeichen ist das "E"

Schritt 3: Das "E" wird mit dem Zeichen an der zweiten Stelle getauscht. Somit wandert das "E" an Stelle 2, das "H" von Stelle 2 an Stelle 7. Das nächste Zeichen ist das "H"

Schritt 4: Das "H" wird mit dem Zeichen an der dritten Stelle getauscht. Somit wandert das "H" an Stelle 3, das "N" von Stelle 3 an Stelle 7. Das nächste Zeichen ist das "N"

Schritt 5: Als Favorit finden wir das erste "N" an Stelle 4. Da es auch an Stelle 4 kommen müsste, verändert sich in diesem Schritt nichts. Das nächste Zeichen ist das "N"

Schritt 6: Das "N" wird mit dem Zeichen an der fünften Stelle getauscht. Somit wandert das "N" an Stelle 5, das "O" von Stelle 3 an Stelle 7. Das nächste Zeichen ist das "O"

Schritt 7: Das "O" wird mit dem Zeichen an der sechsten Stelle getauscht. Somit wandert das "O" an Stelle 6, das "V" von Stelle 6 an Stelle 7. Das nächste Zeichen ist das "R"

Schritt 8: Das "R" wird mit dem Zeichen an der siebten Stelle getauscht. Somit wandert das "R" an Stelle 7, das "V" von Stelle 7 an Stelle 8. Da Stelle 8 die letzte Stelle ist, braucht keine Überprüfung mehr stattzufinden, die Zeichenfolge ist sortiert.

Aufgabe 6

Erzeugen Sie eine Klasse namens Search. Diese besitzt drei Klassenmethoden namens linearSearch, binarySearch und Accesses.

Den Suchmethoden wird ein Feld von Elementen übergeben, die die Schnittstelle Comparable implementieren und ein Element e, dessen Feldindex gesucht wird. Genauer wird der Index eines Elementes o gesucht, so dass $o.compareTo(e) == 0$ gilt.

linearSearch durchsucht das Feld von vorn nach hinten, binarySearch wendet die Binäre Suche an; dazu muss das übergebene Feld sortiert sein.

Die Methode Accesses liefert die Anzahl der Feldzugriffe, die die zuletzt ausgeführte Suchmethode, benötigte (Ziel: Vergleich der Verfahren).

Wenn Ihnen die Realisierung über die Schnittstelle Comparable zu kompliziert erscheint, können Sie zunächst in einem Feld von ganzen Zahlen suchen.

Aufgabe 7

Implementieren Sie Selection Sort. Erzeugen Sie eine Klasse namens Sort, die Sie auf weiteren Übungsblättern erweitern werden. Der Sortiermethode ist eine Klassenmethode, der ein Feld übergeben wird. Die Details sind Ihnen freigestellt: gestalten Sie die Klasse so, dass das Verfahren möglichst universell einsetzbar ist.