Quicksort 1

Ein (meistens) besonders schneller Sortieralgorithmus ist der *Quicksort* Algorithmus. Sein Grundgedanke ist, dass eine Zahl genau dann sortiert ist, wenn alle Zahlen davor kleiner und alle dahinter größer sind. Diese Definition sagt aber nicht aus, dass die Zahlen davor oder danach sortiert sein müssen.

In der Tabelle ist die grau hinterlegte Zahl in diesem Sinne sortiert.

32	21	4	**45**	99	67	87

Der Quick-Sort arbeitet daher umgangssprachlich formuliert so:

a) Wenn nur eine Zahl sortiert werden soll, brich ab.
b) Wähle ein Element aus (das *Pivot-Element*).
c) Schiebe alle Elemente kleiner als das Pivot an eine Stelle vor diesem.
d) Schiebe alle Elemente größer als das Pivot an eine Stelle hinter diesem.
e) Wiederhole von 1. jeweils mit den Elementen vor dem Pivot und denen danach.

Aufgabe 1

Probiere den umgangssprachlichen Algorithmus mit dem bekannten Waage-Programm und mindestens 13 Elementen am Rechner aus.

Macht es einen Unterschied, welches Pivot-Element Du wählst?

Informatik Q1-GK (Ngb) *07.04.2021* *Arbeitsblatt Nr. III.4*

Quicksort 2

Quicksort wird in zwei Teilen implementiert:

a) Eine Methode `quicksort`, die rekursiv aufgerufen wird,
b) eine Methode `teile`.

Für die Methode `quicksort` kann man folgenden *Pseudocode* notieren:

```
quicksort( zahlen, links, rechts)
    wenn links < rechts dann
        teiler = teile(zahlen, links, rechts)

        quicksort(zahlen, links, teiler-1)
        quicksort(zahlen, teiler+1, rechts)
```

Aufgabe 2

Erkläre den abgedruckten *Pseudocode* und was der zweite Teil `teilen` vermutlich für eine Aufgabe hat.

Quicksort 3

Der zweite Teil des Quicksort Algorithmus ist die `teilen` Methode. Sie wählt ein *Pivot-Element* und „schiebt“ alle kleinere Elemente nach „links“ und alle größeren nach „rechts“.

Wie der Algorithmus das „verschieben“ der Elemente effizient schafft, ist eine der genialen Ideen des *Quicksort*.

Aufgabe 3

Holt euch von vorne Sortierzahlen, Marker und ein Pseudocode-Puzzle.

Bringt den Pseudocode in die passende Reihenfolge. Benutzt die Zahlen und Marker zum Ausprobieren.

Analysiert die Methode dann nach diesen Leitfragen:

1. Wie funktioniert die `teilen` Methode?
2. Welches Pivot-Element wird gewählt?
3. Wie werden die Elemente in die richtige „Region“ getauscht?
4. Wie kommt das Pivot-Element am Ende an die richtige Stelle?

Informatik Q1-GK (Ngb) *07.04.2021* *Arbeitsblatt Nr. III.4*

Quicksort 4

Wähle eine der Aufgaben.

Aufgabe 4 Laufzeitmessung

Kopiere das Projekt 🗀`03-Sortiermaschine_3` aus dem Tauschordner. Es enthält Implementierungen aller Sortieralgorithmen (inklusive Quicksort), sowie der bekannten Messmethode.

Ergänze Messungen für den *Quicksort* Algorihtmus zu Deinem Calc-Dokument und vergleiche sie mit denen der anderen Sortieralgorithmen.

Welche obere Grenze für die Laufzeit vermutete Du hier?

Aufgabe 5 Implementierung für gute Programmierer

Implementiere den *Quicksort* in deinem Sortiermaschine-Projekt.

```
1 public int[] quicksortArray( int[] zahlen, int links, int rechts ) {
2 }
```

Du kannst die Testklasse aus dem Projekt 🗀`03-Sortiermaschine_3` im Tauschordner in dein Projekt kopieren. Sie enthält eine Testmethode für den Quicksort.