

## Abituraufgaben zum Thema Laufzeiten von Algorithmen:

2019 III Aufgabe 4

2016 IV Aufgabe 2

2015 IV Aufgabe 2

### Aufgabe : Laufzeit von Selection Sort

```
public class SelectionSort{

    int [] liste = {4,7,2,9,5,1};

    void selectionsort () {

        for (int i = 0;i < liste.length-1;i = i + 1) {

            int position_minelement;

            /* Suche des kleinsten Elements der Restliste */

            position_minelement = suche_position_minelement(i);

            /* Vertausche minelement und i-tes Element */

            tausche_element(position_minelement, i);

        }

    }

    /* Hilfsmethode zur Bestimmung der Position des kleinsten Elements des uebergebenen
    Feldes ab dem i-ten Feldelement */

    int suche_position_minelement (int i) {

        int position_minelement = i;

        for (int j = i;j < liste.length;j = j + 1){

            if (liste[position_minelement] > liste[j]){

                position_minelement = j;

            }

        }

        return position_minelement;

    }

    /* Hilfsmethode zum Vertauschen zweier Elemente */

    void tausche_element (int a, int b) {

        int hilfsvariable = liste[a];

        liste[a] = liste[b];





        liste[b] = hilfsvariable;

    }

}
```

- a) Veranschauliche am Beispiel [4, 7, 2, 9, 5, 1] wie der Sortieralgorithmus funktioniert. Auf Details der Methoden `suche_position_minelement` und `tausche_element` brauchst du beim Veranschaulichen nicht einzugehen.
- b) Wie oft wird bei einer Liste der Länge n die for-Schleife in der Methode `selectionsort` durchlaufen?  
Wie viele Vergleiche sind in der Methode `suche_position_minelement` maximal notwendig?  
Welches Laufzeitverhalten folgt daraus für den Sortieralgorithmus?
- c) Welchen Einfluss haben best case und worst case für diesen Algorithmus?

### Aufgabe: Verschlüsselungsverfahren

- a) Ein sehr einfaches Verschlüsselungsverfahren für Wörter des Alphabets {A,B,C, ...,Z} ist der sog. Cäsar-Code.
- Dabei wird eine Zufallszahl k zwischen 1 und 25 erzeugt und jeder Buchstabe eines Wortes durch den Buchstaben ersetzt, der im Alphabet k Stellen weiter ist. Erreicht man das Ende des Alphabets, geht man zyklisch zum Anfang weiter (Bsp.: k = 3; A → D, Y → B ).
- Schätze ab, wie lange ein Programm zur Entschlüsselung eines zehnstelligen Wortes mit Hilfe des Brute Force Verfahrens benötigt, wenn man für einen Rechenschritt 10 ns veranschlagt.
- b) Bei einer sog. monoalphabetischen Verschlüsselung, wird jeder Buchstabe durch einen Buchstaben eines anderen Alphabets mit gleicher Zeichenzahl ersetzt.  
(z.B. A → , B → , C → , → , ... )
- Schätze ab, wie lange ein Programm für die Entschlüsselung des Codes mit Hilfe des Brute Force Verfahrens benötigt.
- Wie könnte die Entschlüsselungszeit für korrekte Wörter einer Sprache verkürzt werden?

### Aufgabe: Maximumbestimmung

Schreibe ein Java-Methode, mit der von einem gegebenen Zahlenfeld das Maximum bestimmt werden kann.

Untersuche das Laufzeitverhalten in Abhängigkeit der Feldlänge für den best case und den worst case.