

# ESOP Tutorium - Abschlussklausur

---

## 1. String, Rekursion

Implementieren Sie die rekursive Methode `public String reverse (String s)`, welche den übergebenen `String s` umgedreht zurückgibt.

Beispiel:

```
String result = reverse("Heute ist ein spannender Tag");  
System.out.println(result);
```

Ergebnis:

```
gaT rednennaps nie tsi etueH
```

## 2. String, 2D-Array

Implementieren Sie eine Methode `stringSplit(String input)` welche den Eingabestring in ein 2D-Array umwandelt. Der Parameter `input` ist dabei eine kommaseparierte Liste von Vor- und Nachnamen. Sie dürfen davon ausgehen, dass nur ein Vor- und ein Nachname pro Person enthalten ist.

Für jede Person wird ein zweielementiges Array erstellt. Diese Arrays wiederum befinden sich in einem gemeinsamen Array. Das Resultat ist also ein 2D-Array. (siehe Beispiel)

Beispiel:

```
String input = "Andreas Kogler, Fabian Orazé, Timo Tabertshofer";  
System.out.println(Arrays.deepToString(stringSplit(input)));
```

Ergebnis:

```
[[Andreas, Kogler], [Fabian, Orazé], [Timo, Tabertshofer]]
```

## 3. Regex

Gegeben ist folgendes Regex-Pattern in Java. Geben Sie für jede Zeile an, ob `true` oder `false` ausgegeben wird.

```
String pattern = "[,0-9;]*";
```

```
System.out.println("8".matches(pattern));
System.out.println("").matches(pattern);
System.out.println("4-8".matches(pattern));
System.out.println(",1-2;".matches(pattern));
System.out.println(",.;;".matches(pattern));
System.out.println(",12;".matches(pattern));
System.out.println(";;;;;".matches(pattern));
System.out.println("AB".matches(pattern));
System.out.println("493455".matches(pattern));
```

## 4. StringBuilder, char/int

Implementieren Sie eine Methode `public static String encryptCaesar(String clearText, int offset)` die eine Zeichenkette mittels Caesar-Chiffre verschlüsselt.

Dabei wird jedes Zeichen des Eingabetextes (`clearText`) durch ein anderes Zeichen aus dem Alphabet ersetzt. Das `offset` gibt an um wie viele Stellen im Alphabet nach hinten gesprungen wird, um das Ersetzungszeichen zu erhalten. Ein Buchstabe 'a' mit einem `offset` von 2 würde mit einem 'c' ersetzt werden.

Sie dürfen davon ausgehen dass alle Wörter in Kleinbuchstaben angegeben sind. Alle anderen Zeichen (zB Leer- / Satzzeichen) bleiben unverändert.

Beispiel:

```
String alphabet = "abcdefgh";
System.out.println(encryptCaesar(alphabet, 2));

String clearText = "rust is better than java!";
System.out.println(encryptCaesar(clearText, 4));
```

Ergebnis:

```
cdefghi
vywx mw fixxiv xler neze!
```

## 5. Basics - Klassen

Entwerfen Sie eine Klasse `Car`, welche den Verbrauch pro hundert Kilometer, eine Farbe und den Modellnamen eines Autos speichert. Stellen Sie sicher, dass der Verbrauch als `double` gespeichert wird. Alle Variablen sollten nur über Getter- und Setter-Methoden les-/schreibbar sein.

Erstellen Sie einen Konstruktor der alle Instanzvariablen initialisiert. Stellen Sie außerdem sicher dass auch `Car`-Objekte ohne Angabe von Instanzvariablen erstellt werden können.

Implementieren Sie abschließend eine `toString()`-Methode, welche alle Variablen in folgender Form als String retourniert:

"MODELLNAME | FARBE | VERBRAUCH"

Der Verbrauch sollte dabei bei zwei Kommastellen abgeschnitten werden.

Beispiel:

```
Car c1 = new Car();
c1.setModel("Ford Fiesta");

Car c2 = new Car("VW Passat", "Black", 8.2351);

System.out.println(c2);
System.out.println(c1);
```

Ergebnis:

```
VW Passat | Black | 8.23
Ford Fiesta | Null | 0.00
```

## 6. Exceptions

Vervollständigen Sie die `public static calculateBMI(double height, double weight) throws NoHumanSizeException` indem Sie die Funktionalität zum Berechnen vom BMI implementieren. Die Formel dafür lautet:  $\text{Gewicht in kg} / (\text{Größe in m})^2$ .

Fügen Sie zusätzlich Exception Handling für folgendes Szenario ein:

- Wenn die übergebene Größe kleiner als 0 und größer als 2.57 oder das Gewicht kleiner als 0 oder größer als 500 ist, soll eine `NoHumanSizeException` geworfen werden. Implementieren Sie hierfür eine passende Exception welche von `IllegalArgumentException` erbt.

Schreiben Sie eine Main Methode in der Sie zwei Double Werte (height, size) mit einem Scanner einlesen und Ihrer Methode übergeben und dann auf die Konsole ausgeben. Stellen Sie zusätzlich sicher, dass etwaige Exceptions aufgefangen werden.

Beispiel:

```
1.87
95

1.5
0

1.69
53
```

4  
299

Ergebnisse:

27.17

FinalExam.ex06.NoHumanSizeException: There is no human with this weight.

18.56

FinalExam.ex06.NoHumanSizeException: There is no human with this size.

## 7. ArrayList, LinkedList

Erstellen Sie eine ArrayList die Objekte vom Typ `Car` (aus Aufgabe 5) speichern kann, sowie eine LinkedList die beliebige Objekte speichern kann.

Erstellen Sie zwei beliebige Objekte vom Typ `Car` und fügen Sie diese zur ArrayList und der LinkedList hinzu. Fügen Sie außerdem einen beliebigen String und einen Integer zur LinkedList hinzu.

Löschen Sie am Ende das erste Element der ArrayList.