



## Aufgabe 1: Konsequenzen der Digitalisierung

Wie stellen Sie sich die Zukunft in der digitalisierten und algorithmisierten Welt vor:

**Utopie:** Menschen werden durch kooperative digitale Technologie allumfassend unterstützt und in ihren Möglichkeiten erweitert (unsichtbares/ allgegenwärtiges Computing on Demand). Cyborgs und posthumane Lebensformen treten den Menschen an die Seite, Roboter und KIs unterstützen partnerschaftlich den Alltag (insb. von Menschen mit körperlichen und psychischen Einschränkungen). Regelbasierte und einfache entscheidungsbezogene Tätigkeiten werden weitgehend von KIs übernommen, dadurch bleibt für die Menschen mehr Raum für beratende, kreative, kommunikative und innovative Tätigkeiten. Generell wird gesamtgesellschaftlich weniger Lohnarbeit benötigt, deshalb gibt es ein Grundeinkommen für alle.

**Dystopie:** Digitalisierung und Algorithmisierung haben in hohem Maß dazu beigetragen, dass die Welt ungleicher und für Schwächere unsicherer geworden ist. Durch die lückenlose Vermessung von Menschen und ihren Lebensräumen ist es durch KI/ ML möglich geworden, Lebenschancen sehr stark an Herkunftsfamilie, -ort, -kultur und Zugang zu Kapital und Bildung zu binden. Generell werden digitale Technologien von Regierungen zu Überwachungs-, Sanktionierungs- und Steuerungszwecken verwendet. KIs sind mittlerweile den Menschen in ihren Entscheidungsfähigkeiten überlegen und haben wichtige Funktionen in Politik, Wirtschaft und Gesellschaft inne.

**Synthese:** Sämtliche technologische Entwicklungen von Digitalisierung und Algorithmisierung haben mehrdeutige Auswirkungen. Generell geht die gesellschaftliche Absorption von technologischen Neuerungen wie KI, Robotik und autonom agierender Technik nicht eindimensional und linear vonstatten, d.h. selbst wenn sich diese Technologien dynamisch weiterentwickeln, gibt es immer wieder Nebenwirkungen und unintendierte Nebenfolgen, auf die menschlich (z.B. politisch) reagiert werden muss.

Name: \_\_\_\_\_

Matr.-Nr: \_\_\_\_\_



## Aufgabe 2: Grammatiken

Gegeben sei eine Grammatik wie folgt ( $\varepsilon$  ist Leersymbol):

G = <Startsymbol, Terminale, Nicht-Terminale, Regeln> mit

Startsymbol = *Start*

Terminale = {*a, g, h, o, r*}

Nicht-Terminale = {*X, Y, Z*}

Regeln = {*Start*  $\rightarrow$  *XY*,  
*Start*  $\rightarrow$  *YX*,  
*X*  $\rightarrow$  *arg*,  
*X*  $\rightarrow$  *aX*,  
*Y*  $\rightarrow$  *Za*,  
*Y*  $\rightarrow$  *ZaY*,  
*Z*  $\rightarrow$  *oh*}

Entscheiden Sie, ob folgende Wörter in der durch die Grammatik definierten Sprache enthalten sind:

- $\varepsilon$  ja / nein
- *argoha* ja / nein
- *abba* ja / nein
- *ohaarg* ja / nein
- *oharg* ja / nein
- *ohaohaoha* ja / nein
- *ohaohaohaaaarg* ja / nein
- *argohaarg* ja / nein
- *kai* ja / nein
- *ohaohaaaaaaarg* ja / nein

Bestimmen Sie die Wörter, die durch die Grammatik akzeptiert werden in der Art  $a^n b^n$ :

$$\textcolor{red}{a^+rg[oha]^+ \cup [oha]^+a^+rg}$$

Name: \_\_\_\_\_

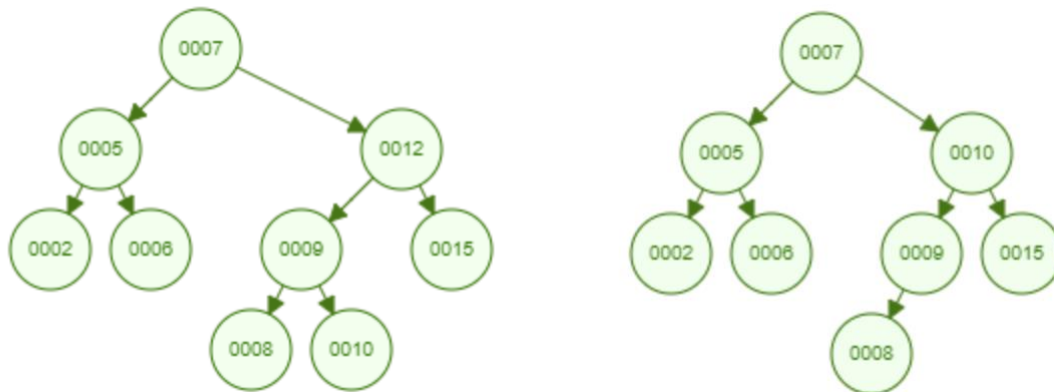
Matr.-Nr: \_\_\_\_\_



## Aufgabe 3: Binärbaum

Gegeben sei eine Implementation eines Binärbaumes (alle kleineren Elemente liegen links vom Element, alle größeren Elemente rechts vom Element), realisiert als verkettete Struktur mit den Zeigern `left` auf den linken Teilbaum, `right` auf den rechten Teilbaum sowie `val` als Variable für den Wert des Teilbaums. Beschreiben Sie die nötigen **Fallunterscheidungen** in einer Funktion `Delete(val)`, die ein Element `Elem` mit dem Wert `val` aus dem Baum löscht (eine Funktion, die den Verweis auf den Knoten mit dem angegebenen Wert `val` zurückliefert, sei gegeben).

Beispiel eines Binärbaumes vor und nach dem Löschen des Wertes 12:



**Fall A:** Zu löschender Knoten hat kein Kind.

*(dann wird beim Löschen einfach der Knoten entfernt.)*

**Fall B:** Zu löschender Knoten hat genau ein Kind.

*(dann wird dieses an die Stelle des zu löschenden Knotens gesetzt.)*

**Fall C:** Zu löschender Knoten hat zwei Kinder.

*(In diesem Fall kann die Löschung sowohl über den linken wie über den rechten Teilbaum vollzogen werden. Um die in-order-Reihenfolge aufrechtzuerhalten, ist aber ein Abstieg bis zu einem Halbblatt unvermeidlich, in dem z.B. der zu löschende Knoten durch einen (unmittelbaren) Nachbarn in der in-order-Reihenfolge ersetzt wird.)*



## Aufgabe 4: Prolog Unifikation

Gegeben seien jeweils 2 Prolog-Ausdrücke. Sie sollen entscheiden, ob diese beiden Ausdrücke unifizierbar sind und, falls dieses der Fall ist, den durch Substitution erzeugten Ausdruck angeben. Sollte die Unifikation misslingen, geben Sie den Grund an. Als Beispiele sind gegeben (Konstanten fangen mit kleinem, Variablen mit großem Buchstaben an):

- `test (X, foo, Y) = test (M, N, M)` ☺ mit `test (X, foo, X)`.
- `test (kai, X) = test1(kai, udo)` ☹ weil `test ≠ test1`.

Hier nun die Ausdrücke:

- `foo(Bar) = bar(Foo)`.

☹ `foo ≠ bar`.

- `[foo, Bar] = [Bar, foo]`.

☺ mit `Bar = foo`.

- `name(Kai) = name(kai, von, luck)`.

☹ `name/1 ≠ name/3`.

- `name(Kai) = name([kai, von, luck])`.

☺ mit `Kai = [kai, von, luck]`.

## Aufgabe 4: Prolog Programm

Was macht das folgende Prolog Programm `foo(List1, List2, Result)` :

```
foo([],_, []).  
foo([Elem|Rest],List,Result) :-  
    not(member(Elem,List)), foo(Rest,List,Result).
```

Name: \_\_\_\_\_

Matr.-Nr: \_\_\_\_\_



```
foo([Elem|Rest],List,[Elem|Result]) :-  
    member(Elem,List), foo(Rest,List,Result).
```

Beschreibung mit wenigen Worten:

`foo(List1,List2,Result)` **bildet den Durchschnitt zweier Listen**

Was ergibt sich bei den Aufrufen:

```
foo([1,2,3,4],[1,2,3,4], Result).
```

```
Result = [1,2,3,4].
```

```
foo([1,2,3,4],[3,4,5,6], Result).
```

```
Result = [3,4].
```

```
foo([1,2,3,4],[5,6,7,8], Result).
```

```
Result = [].
```



## Bewertung der Aufgaben

**Aufgabe 1:** irgendwie zutreffende Beschreibung/Aspekt = 0,5 Punkte  
hinreichend gute Beschreibung/Aspekt = 1 Punkt  
(maximal 3 Punkte)

**Aufgabe 2:** pro richtige Wortentscheidung = 0.5 Punkte  
fast richtiger regulärer Ausdruck = 0,5 Punkte  
richtiger regulärer Ausdruck = 1 Punkt  
(maximal 6 Punkte)

**Aufgabe 3:** richtiger Fall: keine Kinder = 1 Punkt  
richtiger Fall: ein Kind = 1 Punkt  
richtiger Fall: zwei Kinder = 1 Punkt  
(maximal 3 Punkte)

**Aufgabe 4:** jeweils richtiges Ergebnis = 1 Punkt  
(maximal 4 Punkte)

**Aufgabe 5:** fast richtige Idee = 1 Punkt  
richtige Idee = 2 Punkte  
pro richtiges Resultat = 1 Punkt  
(maximal 5 Punkte)

**Maximale Punkte aller Aufgaben:** = 21 Punkte

**Notengebung:** Maximum(Minimum(Aufrunden(Summe aller Punkte - 4),15),0)

Name: \_\_\_\_\_

Matr.-Nr: \_\_\_\_\_