



Aufgabe 1: Konsequenzen der Digitalisierung

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

Utopie:

Dystopie:

Synthese:

Name: _____

Matr.-Nr: _____



Aufgabe 2: Grammatiken

Gegeben sei eine Grammatik wie folgt (ε 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:

- ε 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 $a^n b^n$:

Name: _____

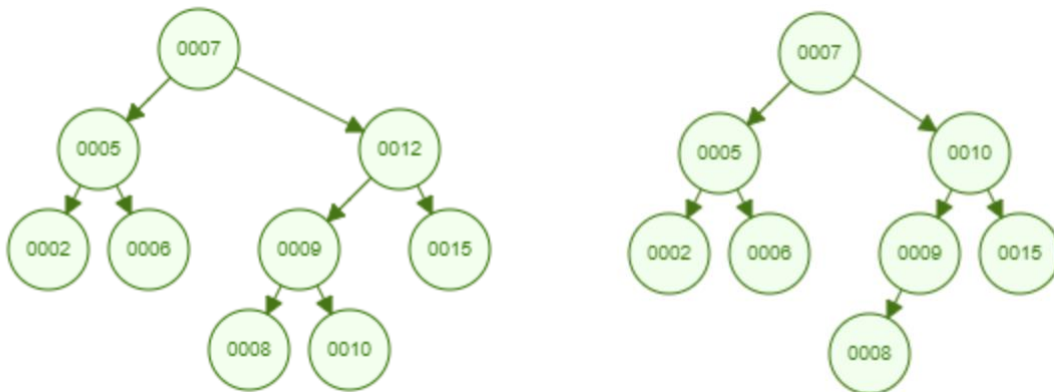
Matr.-Nr: _____



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:





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

Aufgabe 4: Prolog Programm

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

```
foo([],_, []).
```

Name: _____

Matr.-Nr: _____



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

Beschreibung mit wenigen Worten:

```
foo(List1,List2,Result)
```

Was ergibt sich bei den Aufrufen:

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

Result =

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

Result =

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

Result =