

## Aufgabe 1: Digitale Disruption

Nennen Sie eine IT Technologie, die die klassische Medienlandschaft stark verändert hat und skizzieren Sie diese Veränderung (mit maximal 100 Wörtern):

IT Technologie:

Veränderung des Medienbereichs durch diese Technologie:

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

## Aufgabe 2: Grammatiken

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

G  
mit

=  $\langle \text{Startsymbol, Terminale, Nicht-Terminale, Regeln} \rangle$

Startsymbol = *Start*

Terminale =  $\{a, b, c\}$

Nicht-Terminale =  $\{X, Y\}$

Regeln =  $\{ \text{Start} \rightarrow Xb, \\ \text{Start} \rightarrow XbY, \\ X \rightarrow ab, \\ X \rightarrow aXb, \\ Y \rightarrow c, \\ Y \rightarrow cY \}$

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

- |                 |           |
|-----------------|-----------|
| • $\varepsilon$ | ja / nein |
| • aabccb        | ja / nein |
| • abcd          | ja / nein |
| • aabbaacb      | ja / nein |
| • acdc          | ja / nein |
| • aabbbc        | ja / nein |
| • abc           | ja / nein |
| • abbccc        | ja / nein |
| • bc            | ja / nein |
| • abb           | ja / nein |

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

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

## Aufgabe 3: Endliche Automaten

Gegeben sei ein Getränkeautomat, der 20 Cent und 50 Cent nimmt. Werden 60 Cent eingeworfen, kommt eine Flasche Wasser, bei 70 Cent eine Brause, bei mehr als 70 Cent ebenfalls eine Brause und der überzahlte Betrag heraus. Skizzieren Sie den entsprechenden endlichen Automaten.

## Aufgabe 4: Aussagenlogik

Überprüfen Sie den Wahrheitswert der jeweiligen aussagenlogischen Ausdrücke mit Hilfe einer Wahrheitstabelle:

$$\neg(P \wedge Q) \Leftrightarrow ((\neg P \vee \neg Q) \wedge (P \vee \neg P))$$

| P | Q |  |
|---|---|--|
| 0 | 0 |  |
| 0 | 1 |  |
| 1 | 0 |  |
| 1 | 1 |  |

$$(P \wedge Q) \Leftrightarrow ((P \wedge \neg R) \wedge (P \wedge Q)) \vee (P \wedge Q)$$

| P | Q | R |  |
|---|---|---|--|
| 0 | 0 | 0 |  |
| 0 | 0 | 1 |  |
| 0 | 1 | 0 |  |
| 0 | 1 | 1 |  |
| 1 | 0 | 0 |  |
| 1 | 0 | 1 |  |
| 1 | 1 | 0 |  |
| 1 | 1 | 1 |  |

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

## Aufgabe 5: 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):

- $\text{test}(X, \text{foo}, Y) = \text{test}(M, N, M)$  ☺ mit  $\text{test}(X, \text{foo}, X)$ .
- $\text{test}(\text{kai}, X) = \text{test1}(\text{kai}, \text{udo})$  ☹ weil  $\text{test} \neq \text{test1}$ .

Hier nun die Ausdrücke:

- $\text{coautor}(\text{Koenig}, \text{roland}) = \text{coautor}(\text{esther}, \text{Seifert})$ .
- $\text{member}(X, [2, 4, 6]) = \text{member}(4, [2, X, 6])$ .
- $\text{member}(X, [2, 4, 6]) = \text{member}(4, X)$ .
- $\text{verwandt}(\text{paul}, \text{paula}) = \text{verwandt}(\text{paula}, \text{paul})$ .

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

## Aufgabe 6: Prolog Programmanalyse

Gegeben ein Prolog-Programm `foo(List1, List2)` wie folgt

```
foo([], []).  
foo([a|Bar],[x|XBar]) :- foo(Bar,XBar).  
foo([_|Bar],XBar) :- foo(Bar,XBar).
```

Beschreiben Sie kurz, was das Programm macht:

Zu was unifiziert `List` in den folgende Beispielen:

`foo([a, b, b, a], List).`

`List =` .

`foo([a, c, d, c], List).`

`List =`

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

`List =`