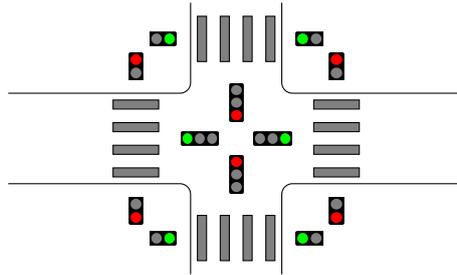


3.0 VU Formale Modellierung			
185.A06		WS 2015/SS 2016	24. Mai 2016
Matrikelnummer	Familienname	Vorname	Gruppe <b>A</b>

**Aufgabe 1 (10 Punkte)** Eine kreuzförmige Straßenkreuzung ist mit Fahrzeugampeln (vier Phasen) und Fußgängerampeln (zwei Phasen) für alle Richtungen ausgestattet. Es gibt keine speziellen Ampeln für Abbieger.



Modellieren Sie die Schaltfolgen der Ampeln mit Hilfe eines Petri-Netzes. Geben Sie eine geeignete Anfangsmarkierung an. Geben Sie den Stellen und Transitionen geeignete Bezeichnungen, die ihre Rolle beschreiben.

Hinweis: Ampeln, die sich identisch verhalten, können als eine einzige Ampel betrachtet werden.

**Aufgabe 2 (10 Punkte)** DATALOG ist eine logikorientierte Datenbanksprache. DATALOG-Programme bestehen aus Folgen von sogenannten Klauseln. Es gibt zwei Arten von Klauseln: Fakten und Regeln.

Ein Faktum besteht aus einer Atomformel gefolgt von einem Punkt. Eine Regel besteht aus einer Atomformel, gefolgt von den Zeichen `:-` sowie einer nicht-leeren Liste von Atomformeln, die durch Beistriche getrennt werden. Regeln enden ebenfalls mit einem Punkt.

Eine Atomformel ist ein Name, dem optional eine in runden Klammern eingeschlossene Argumentliste folgen kann. Eine Argumentliste ist eine nicht-leere Folge von Namen und Variablen in beliebiger Reihenfolge, die voneinander durch Kommas getrennt werden.

Ein Name ist eine nicht-leere Folge von Buchstaben und Ziffern, die mit einem Kleinbuchstaben beginnt. Eine Variable ist eine nicht-leere Folge von Buchstaben und Ziffern, die mit einem Großbuchstaben beginnt.

Beispielsweise besteht das folgende DATALOG-Programm aus zwei Fakten und drei Regeln; `adam`, `seth`, `istKindVon` usw. sind Namen, `X` und `Y` sind Variablen.

```

istKindVon(seth,adam).
istKindVon(enosh,seth).
istNachfahreVon(X,Y) :- istKindVon(X,Y).
istNachfahreVon(X,Z) :- istKindVon(X,Y), istNachfahreVon(Y,Z).
istMensch(X) :- istNachfahreVon(X,adam).

```

Beschreiben Sie die zulässigen DATALOG-Programme mittels einer kontextfreien Grammatik. Verwenden Sie so weit wie möglich EBNF-Notationen, um die Grammatik übersichtlich zu halten und rekursive Regeln zu vermeiden.

**Aufgabe 3 (10 Punkte)** Thomas wird im Sommer in das Ferienlager am Wolfgangsee fahren. Um den Organisatoren die Planung zu erleichtern, soll er davor bekannt geben, welche Sportarten (maximal drei) er dort ausüben möchte. Thomas kann sich nur schwer entscheiden und schreibt folgende Email.

*Liebe Organisatoren,  
ich freue mich schon sehr auf den Sommer und das Camp!  
Fußball und Handball sind beide ok, ich möchte aber auf keinen Fall beide machen, das wäre mir zu anstrengend. Mein Freund Florian kommt auch ins Camp und geht Bogenschießen, das möchte ich auf jeden Fall auch machen! Falls ich beim Rudern teilnehme, möchte ich auch segeln, damit sich der Weg zum See auszahlt; Segeln ohne Rudern kann ich mir aber schon vorstellen, das ist alleine auch spannend genug. Ich möchte gerne meine Oberarme trainieren, dafür sind wohl Rudern oder Handball ganz gut geeignet; auch beide gleichzeitig wären in Ordnung.  
Bis bald, Thomas*

- Helfen Sie den Organisatoren die Email zu analysieren, indem Sie die beschriebenen Wünsche mit allen Anhaltspunkte durch aussagenlogische Formeln ausdrücken. Geben Sie die Bedeutung der von Ihnen verwendeten Aussagenvariablen an.
- Lassen sich Thomas' Wünsche berücksichtigen? Welche Varianten sind möglich? Begründen Sie die Antwort mit Hilfe Ihrer aussagenlogischen Modellierung.

**Aufgabe 4 (10 Punkte)** Seien *Beherbergt*/2, *Gefährlich*/1, *Tier*/1 und *Zoo*/1 Prädikaten-symbole sowie *elefant*, *löwe* und *schlange* Konstantensymbole mit folgender Bedeutung:

<i>Beherbergt</i> ( $x, y$ )	... $x$ beherbergt $y$	<i>elefant</i>	... Elefant
<i>Gefährlich</i> ( $x$ )	... $x$ ist gefährlich	<i>löwe</i>	... Löwe
<i>Tier</i> ( $x$ )	... $x$ ist ein Tier	<i>schlange</i>	... Schlange
<i>Zoo</i> ( $x$ )	... $x$ ist ein Zoo		

Verwenden Sie diese Symbole, um die beiden nachfolgenden Sätze in prädikatenlogische Formeln zu übersetzen.

- Alle Zoos beherbergen Schlangen, Löwen oder beides.
- Es gibt einen Zoo, der alle gefährlichen Tiere beherbergt.

Sei weiters folgende Interpretation gegeben:

$$\mathcal{U} = \{\text{Affe, Berlin, Elefant, Krokodil, Löwe, München, Pinguin, Robbe, Salzburg, Schlange, Tiger, Wien}\}$$

$$I(\text{Beherbergt}) = \{(\text{Berlin, Affe}), (\text{Berlin, Löwe}), (\text{Berlin, Robbe}), (\text{München, Elefant}), (\text{München, Löwe}), (\text{Salzburg, Löwe}), (\text{Salzburg, Robbe}), (\text{Wien, Affe}), (\text{Wien, Krokodil}), (\text{Wien, Schlange}), (\text{Wien, Tiger})\}$$

$$I(\text{Gefährlich}) = \{\text{Affe, Krokodil, Löwe, Robbe}\}$$

$$I(\text{Tier}) = \{\text{Löwe, Pinguin, Robbe, Schlange, Tiger}\}$$

$$I(\text{Zoo}) = \{\text{Berlin, Salzburg, Wien}\}$$

$$I(\text{löwe}) = \text{Löwe} \quad I(\text{schlange}) = \text{Schlange} \quad I(\text{elefant}) = \text{Elefant}$$

Übersetzen Sie die nachfolgenden Formeln in natürliche Sprache. Geben Sie an, ob die Formeln in der angegebenen Interpretation  $I$  wahr oder falsch sind. Begründen Sie Ihre Antwort; es ist keine formale Auswertung erforderlich.

- c)  $\exists x \exists y (Zoo(x) \wedge Gefährlich(y) \wedge Beherbergt(x, y))$   
d)  $\forall x (Beherbergt(x, schlange) \neq Beherbergt(x, löwe))$   
e)  $\exists x (Zoo(x) \wedge Beherbergt(x, elefant) \wedge Beherbergt(x, löwe))$   
f)  $\forall x \exists y (Zoo(x) \wedge Gefährlich(y) \wedge Beherbergt(x, y))$

**Aufgabe 5 (10 Punkte)** In Büchern über formale Sprachen ist folgende Definition zu finden:

Eine *reguläre Grammatik* wird durch ein 4-Tupel  $G = \langle V, T, P, S \rangle$  festgelegt, wobei

- $V$  und  $T$  endliche, disjunkte Mengen von Symbolen sind ( $V \cap T = \{\}$ ),
- $S$  ein Symbol aus  $V$  ist ( $S \in V$ ) und
- $P \subseteq V \times (T \cdot V \cup \{\varepsilon\})$  eine endliche Menge von Paaren ist.

Die Elemente von  $P$  werden Produktionen genannt; statt  $(x, y) \in P$  wird auch  $x \rightarrow y$  geschrieben. Die Notation  $x \rightarrow y_1 \mid \dots \mid y_n$  ist eine Abkürzung für die Produktionen  $x \rightarrow y_1, \dots, x \rightarrow y_n$ .

Das Wort  $u y v$  ist aus dem Wort  $u x v$  in einem Schritt ableitbar, geschrieben  $u x v \Rightarrow u y v$ , wenn  $x \rightarrow y$  gilt. Die von  $G$  generierte Sprache  $\mathcal{L}(G)$  ist definiert als die Menge  $\{w \in T^* \mid S \xRightarrow{*} w\}$ , wobei  $\xRightarrow{*}$  den reflexiven und transitiven Abschluss von  $\Rightarrow$  bezeichnet.<sup>1</sup>

Geben Sie an, welche der folgenden Tupeln eine reguläre Grammatik gemäß der obigen Definition darstellt. Begründen Sie Ihre Antwort, falls es sich um keine reguläre Grammatik handelt. Entspricht das Tupel der Definition, geben Sie die Sprache an, die durch die Grammatik generiert wird.

- a)  $\langle \{X\}, \{a, b\}, \{X \rightarrow aXb \mid \varepsilon\}, X \rangle$   
b)  $\langle \{X, Y\}, \{a, b\}, \{X \rightarrow aX \mid bY \mid \varepsilon, Y \rightarrow bY\}, X \rangle$   
c)  $\langle \{a, b\}, \{X, Y\}, \{a \rightarrow Xa \mid Yb \mid \varepsilon, b \rightarrow Yb\}, b \rangle$   
d)  $\langle \{X, Y\}, \{a, b\}, \{X \rightarrow Xa \mid Yb \mid \varepsilon, Y \rightarrow Yb \mid \varepsilon\}, X \rangle$

<sup>1</sup>Das heißt, dass  $\xRightarrow{*}$  die kleinste Relation mit folgenden Eigenschaften ist:

- Aus  $u \Rightarrow v$  folgt  $u \xRightarrow{*} v$ .
- Es gilt  $u \xRightarrow{*} u$  für alle Wörter  $u \in T^*$ .
- Aus  $u \xRightarrow{*} v$  und  $v \xRightarrow{*} w$  folgt  $u \xRightarrow{*} w$ .

Anschaulich gesprochen steht  $\xRightarrow{*}$  für die Ableitbarkeit in beliebig vielen Schritten.