1. Welche der folgenden Aussagen zur Berechenbarkeitstheorie ist richtig?

A. Jedes lösbare Problem ist auch algorithmisch lösbar.

B. Es ist möglich ein Turingmaschinenprogramm zu schreiben, welches testet ob eine beliebige Grammatik mehrdeutig ist.

C. Die Church-Turing These impliziert, dass jede bekannte Programmiersprache Turing-vollständig ist.

D. Turingmaschinen mit einem zweiseitig unbeschränkten Band sind echt mächtiger als TMs mit einem einseitig beschränkten Band.

E. Turingmaschinen können nur mit einem einseitig unbeschränkten Band definiert werden.

F. Die Church-Turing These impliziert, dass jeder Algorithmus auch als Registermaschinenprogramm ausgedrückt werden kann.

---

2. Betrachten Sie die formalen Sprachen $L = \{\epsilon, \mathsf{ca}, \mathsf{aba}\}$, $M = \{\mathsf{a}, \mathsf{b}, \mathsf{c}\}$ und $N = \{\mathsf{a}, \mathsf{b}, \mathsf{c}\}^*$. Was ist $(LM) \cap N$?

A. $(LM) \cap N = \{\mathsf{a}, \mathsf{b}, \mathsf{c}\}^*$

B. $(LM) \cap N = \varnothing$

C. $(LM) \cap N = \{\epsilon, \mathsf{a}, \mathsf{b}, \mathsf{c}, \mathsf{caa}, \mathsf{cac}, \mathsf{abac}\}$

D. $(LM) \cap N = \{\mathsf{caa}, \mathsf{cab}, \mathsf{cac}, \mathsf{abaa}, \mathsf{abab}, \mathsf{abac}\}$

E. $(LM) \cap N = \{\epsilon, \mathsf{a}, \mathsf{b}, \mathsf{c}, \mathsf{caa}, \mathsf{cab}, \mathsf{cac}, \mathsf{abaa}, \mathsf{abab}, \mathsf{abac}\}$

F. $(LM) \cap N = \{\mathsf{a}, \mathsf{b}, \mathsf{c}, \mathsf{caa}, \mathsf{cab}, \mathsf{cac}, \mathsf{abaa}, \mathsf{abab}, \mathsf{abac}\}$

---

3. Welche der folgenden Anordnungen der Chomsky-Hierarchie ist falsch? Zur Erinnerung:

$\mathcal{L}_3 =$ reguläre Sprachen
$\mathcal{L}_2 =$ kontextfreie Sprachen
$\mathcal{L}_1 =$ kontextsensitive Sprachen
$\mathcal{L}_0 =$ rekursiv aufzählbare Sprachen
$\mathcal{L} =$ formale Sprachen

A. $\mathcal{L}_3 \subseteq \mathcal{L}_2 \subsetneq \mathcal{L}_1 \subseteq \mathcal{L}_0 \subsetneq \mathcal{L}$

B. $\mathcal{L}_3 \neq \mathcal{L}_2 \neq \mathcal{L}_1 \neq \mathcal{L}_0 \neq \mathcal{L}$

C. $\mathcal{L}_3 \subseteq \mathcal{L}_2 \subseteq \mathcal{L}_1 \subseteq \mathcal{L}_0 \subseteq \mathcal{L}$

D. $\mathcal{L}_3 \subsetneq \mathcal{L}_2 \subsetneq \mathcal{L}_1 \subsetneq \mathcal{L}_0 \subsetneq \mathcal{L}$

E. $\mathcal{L}_3 = \mathcal{L}_2 = \mathcal{L}_1 = \mathcal{L}_0 = \mathcal{L}$

---

4. Welche der folgenden Aussagen zur Booleschen Algebra ist richtig?

A. Sei $\mathcal{B}$ eine Boolesche Algebra und sei B die Trägermenge von $\mathcal{B}$. Für alle $a, b \in B$ gilt: $\overline{a + b} = \overline{a} + \overline{b}$.

B. Jeder Boolesche Ausdruck kann in eine disjunktive Normalform umgewandelt werden, nicht jedoch in eine konjunktive Normalform.

C. Die binäre Algebra ist eine Algebra, aber keine Boolesche Algebra.

D. Die Mengenalgebra ist eine Boolesche Algebra, aber nicht jede Boolesche Algebra ist isomorph zu einer Mengenalgebra.

E. Jede Boolesche Algebra ist isomorph zu einer Mengenalgebra.

---

5. Welche der folgenden Aussagen zu Normalformen einer aussagenlogischen Formeln A ist richtig?

A. Jede disjunktive Normalform von A ist eine Disjunktion von Konjunktionen von Atomen.

B. Jede konjunktive Normalform von A ist eine Disjunktion von Konjunktionen von Literalen.

C. Jede Formel A besitzt eine disjunktive, nicht jedoch eine konjunktive Normalform.

D. Jede Formel A besitzt eine konjunktive, nicht jedoch eine disjunktive Normalform.

E. Jede Formel A besitzt sowohl eine konjunktive wie eine disjunktive Normalform.

---

6. Welche der folgenden Aussagen zur Aussagenlogik ist richtig?

A. Es existiert eine Wahrheitsfunktion f, sodass f nicht durch eine Wahrheitstabelle ausgedrückt werden kann.

B. Es gelten die Gesetze von de Morgan, das heißt: $\neg(E \land F) \approx \neg E \land \neg F$

C. Es gilt die logische Äquivalenz: $A \lor \mathsf{True} \approx A$

D. Eine Formel A ist eine Tautologie gdw. $\neg A$ erfüllbar ist.

E. Die Aussagenlogik ist vollständig axiomatisierbar, das heißt es existiert ein korrektes und vollständiges Beweissystem für die Aussagenlogik.

---

7. Algebra: Formen Sie die Ausdrücke 1, 2 und 3 über der Booleschen Algebra

$$\langle\{\mathsf{True}, \mathsf{False}\}; \vee, \wedge, \neg, \mathsf{False}, \mathsf{True}\rangle$$

mit *aus der VO bekannten Äquivalenzen* um, um festzustellen welche der gegebenen Ausdrücke äquivalent sind. Begründen Sie Ihre Entscheidungen.

1. $(\neg b \wedge (\neg c \vee a) \wedge (\neg c \vee b)) \vee c$

2. $((\neg a \wedge b) \vee c) \vee (a \wedge \neg b)$

3. $(a \vee b \vee c) \wedge (\neg b \vee \neg a \vee c)$

`[16 Punkte]`

---

8. Algebra: Betrachten Sie folgenden Gleichungen über der Signatur $F = \{\mathsf{a}, \mathsf{b}, \mathsf{f}, \mathsf{g}, \mathsf{h}\}$, wobei a und b die Stelligkeit 0, f die Stelligkeit 2, sowie g und h die Stelligkeit 1 haben:

$$E := \{\mathsf{f}(\mathsf{a}, x) = \mathsf{g}(x), \mathsf{f}(x, \mathsf{b}) = \mathsf{h}(x)\}$$

Zeigen Sie, dass $E \vdash \mathsf{g}(\mathsf{b}) = \mathsf{h}(\mathsf{a})$ gilt.

Argumentieren Sie informell, übertragen Sie Ihre Argumentation in einen formalen Beweis. Notieren Sie die verwendeten Inferenzregeln der Gleichungslogik.

Sie erhalten die volle Punktezahl, wenn Sie den formalen Beweis mit verwendeten Regeln notieren.

---

9. Formale Sprachen:

a) Es sei die Sprache L über dem Alphabet $\Sigma = \{\mathsf{a}, \mathsf{b}, \mathsf{c}\}$ gegeben:

$$L = \{u\mathsf{c}v \mid u, v \in \Sigma^* \text{ und } \#\mathsf{a}(u) = \#\mathsf{b}(v)\},$$

wobei $\#\mathsf{a}(w)$ die Anzahl von Symbol a in Wort w repräsentiert. Geben Sie eine Grammatik G für die Sprache L an. `[8 Punkte]`

b) Gegeben sei die beschränkte Grammatik $G = (\{S, B\}, \{\mathsf{a}, \mathsf{b}, \mathsf{c}\}, R, S)$, wobei die Regeln R wie folgt definiert sind:

$$\begin{aligned} S &\to \mathsf{a}SB\mathsf{c} \mid \mathsf{abc} \\ \mathsf{c}B &\to B\mathsf{c} \\ \mathsf{b}B &\to \mathsf{bb} \end{aligned}$$

Geben Sie eine Ableitung für den String aabbcc an. `[8 Punkte]`

---

10. Berechenbarkeitstheorie: Schreiben Sie ein `while`-Programm P für eine Registermaschine $R_{swap} = ((x_i)_{1 \leqslant i \leqslant 3}, P)$, welches den Inhalt zweier Register vertauscht.

Am Beginn steht Zahl n in Register x_1, Zahl m in Register x_2 und eine unbekannte Zahl in Register x_3. Am Ende der Ausführung soll n in Register x_2 stehen und m in Register x_1. `[16 Punkte]`

ANSWERKEY FOR "versionG"

```
Version 1:  F  F  E  E  E  E
```