Klausur zur Vorlesung „Berechenbarkeit und Logik“

SS 2024 / 22. Juli 2024

Vorname: ______________________________

Nachname: ______________________________

Matrikelnummer: ______________________________

Aufgabe	**Punktzahl**	**Erreicht**
1	10	
2	5	
3	3	
4	4	
5	5	
6	5	
7	6	
8	6	
9	6	
10	6	
11	6	
12	8	
Bonus	0	
$\boldsymbol{\Sigma}$	70	

Generelle Hinweise:

- Prüfungsdauer: **120 Minuten**
- Wenn Sie in der Klausur **35 Punkte** erreichen, haben Sie mit Sicherheit bestanden.
- Hilfsmittel: Ein beidseitig handbeschriebenes DIN-A4-Blatt.
- Benutzen Sie ein **dokumentenechtes Schreibgerät**.
- Überprüfen Sie die Ihnen ausgehändigte Klausur auf Vollständigkeit (**13 Aufgaben** auf 12 Seiten).
- Tragen Sie **auf jedes Blatt** Ihren **Namen** und Ihre **Matr.-Nr.** in die entsprechenden Felder ein.
- Schreiben Sie Ihre Lösungen in die dafür vorgesehenen Felder. Reicht der Platz in einem Feld nicht aus, so benutzen Sie die Rückseite des entsprechenden Blattes und vermerken Sie dies auf der Vorderseite. Reicht der Platz dennoch nicht aus, können Sie die Aufsicht nach zusätzlichen Blättern fragen.
- Schreiben Sie bitte **deutlich**. Unleserliche Lösungen sind ungültig.
- Ein **Täuschungsversuch** führt umgehend zum Ausschluss und **Nichtbestehen**. Es erfolgt keine Vorwarnung.
- Alle mitgeführten **elektronischen Geräte** sind vor der Klausur bzw. spätestens jetzt auszuschalten.

Inhaltliche Hinweise:

- In den WHILE- und LOOP-Programmen dürfen Sie die Addition, Multiplikation, die in der Vorlesung definierte Subtraktion und für eine Variable x die Bedingung IF $x = 0$ THEN P END als WHILE- bzw. LOOP-berechenbar voraussetzen und in Ihren Programmen verwenden. Geben Sie an, in welcher Variable der Ausgabewert am Ende steht.
- Sie dürfen annehmen, dass die Addition und Multiplikation zweier natürlicher Zahlen primitiv-rekursiv sind.
- Für die Konstruktion von μ- bzw. primitiv-rekursiven Funktionen und für WHILE- und LOOP-Programme darf die Schreibweise aus den Übungen benutzt werden.

Name: **Matrikelnummer:**

Aufgabe 1. (10 Punkte) Beantworten Sie die folgenden fünf Fragen. Für jede Teilaufgabe gibt es zwei Punkte, wenn Sie die Aufgabe vollständig und korrekt beantwortet haben.

(a) Wie viele paarweise nicht äquivalente aussagenlogische Formeln mit n atomaren Formeln gibt es?
Wie viele sind es, wenn man sich auf Formeln in DNF beschränkt?

Lösung: 2^{2^n} Stück (für beide Fragen). Es gibt zu jeder Formel eine äquivalente Formel in DNF.

(2) Sortieren Sie die folgenden 4 Mengen von Funktionen $f : \mathbb{N} \to \mathbb{N}$ so, dass die i-te Menge Ihrer Liste in der $(i+1)$-ten enthalten ist.
(a) $\{f \mid f \text{ ist total und } \mu - \text{rekursiv}\}$ (b) $\{f \mid f \text{ ist eine quadratische Funktion}\}$
(c) $\{f \mid f \text{ ist Turing-berechenbar}\}$ (d) $\{f \mid f \text{ ist Loop-berechenbar}\}$

Lösung: b, d, a, c

(3) Geben Sie drei verschiedene nicht entscheidbare Probleme an.

Lösung: Halteproblem, PCP, Schnittproblem von CFGs

(4) Hat folgende PCP-Instanz eine Lösung? Begründen Sie!

$$\begin{pmatrix} a & ab & bac \\ aa & ba & bb \end{pmatrix}.$$

Lösung: Es gibt keine Lösung. Die letzte Spalte kann man nicht benutzen, da x_3 der einzige Eintrag mit einem c ist. Somit muss eine gültige PCP-Lösung mit (x_1, y_1) beginnen. Damit haben wir aber stets mindestens ein Zeichen mehr in der zweiten Komponente als in der ersten Komponente, was nicht sein darf.

(5) Es sei $\mathcal{F}$ die Menge aller Formeln der Prädikatenlogik. Geben Sie eine (beliebige) Teilmenge $\mathcal{F}' \subseteq \mathcal{F}$ an, die entscheidbar ist.

Lösung: Zum Beispiel $\mathcal{F}' = \{F \in \mathcal{F} \mid F \text{ beginnt mit einem Allquantor}\}$

Aufgabe 2. (5 Punkte) Geben Sie ein LOOP-Programm an, das die folgende Funktion $f\colon \mathbb{N}^3 \to \mathbb{N}$ berechnet:

$$f(x,y,z) = \begin{cases} x^{\min(y,z^3)}, & \text{falls } x \leq \min(y,z^3), \\ x + \min(y,z^3), & \text{sonst.} \end{cases}$$

Kommentieren Sie neben Ihrer Lösung kurz, was Ihre Teilprogramme ausgeben (Beispiel: [Ihr Code für $\min(x,y)$] - „Berechnung von $\min(x,y)$“).

Lösung: Eingabe x_1, x_2, x_3, Ausgabe in x_1

$x_4 := x_3 \cdot x_3;$
$x_4 := x_4 \cdot x_3;$
$x_5 := x_2 - x_4;$
IF $x_5 = 0$ THEN $x_4 := x_2$ END;
$x_6 := x_1 - x_4;$
$x_7 := x_1 + x_4;$
IF $x_6 = 0$ THEN
  $x_7 := 1;$
  LOOP x_4 DO $x_7 := x_1 \cdot x_7$ END
END;
$x_1 := x_7$

Erläuterungen: In x_4 speichern wir $\min(y,z^3)$. In x_7 wird zunächst $x + \min(y,z^3)$ gespeichert. Wenn $x \leq \min(y,z^3)$, wird x_7 mit $x^{\min(y,z^3)}$ überschrieben.

Aufgabe 3. (3 Punkte) Geben Sie an, welche partielle Funktion $f : \mathbb{N}^2 \to \mathbb{N}$ das folgende GoTo-Programm berechnet (Eingabevariablen sind x_1 und x_2, die Ausgabevariable ist x_1):

$$
\begin{array}{ll}
M_1: & x_3 := x_2 + x_1\,; \\
 & x_4 := x_3 - 5\,; \\
 & \textsc{If}\ x_4 = 0\ \textsc{Then}\ \textsc{GoTo}\ M_2\,; \\
 & \textsc{GoTo}\ M_1\,; \\
M_2: & x_1 := x_1 + x_3\,; \\
 & \textsc{If}\ x_3 = 0\ \textsc{Then}\ \textsc{GoTo}\ M_3\,; \\
 & \textsc{Halt} \\
M_3: & x_1 := x_2 + 5\,; \\
 & \textsc{Halt}
\end{array}
$$

Lösung:

$$f(x,y) = \begin{cases} 2x + y, & \text{wenn } 0 < x + y \leq 5 \\ 5 & \text{wenn } x, y = 0 \\ \text{undefiniert}, & \text{sonst.} \end{cases}$$

Aufgabe 4. (4 Punkte) Geben Sie an, welche Funktionen von μf und μg berechnet werden, wobei f und g wie folgt definiert sind.

(a) $f\colon \mathbb{N}^4 \to \mathbb{N}$ mit $f(n,x,y,z) = n \cdot x + n \cdot y^2 + 2z$

Lösung: $\mu f(x,y,z) = \begin{cases} 0 & \text{falls } z = 0, \\ \text{undefiniert}, & \text{sonst.} \end{cases}$

(b) $g\colon \mathbb{N}^3 \to \mathbb{N}$ mit $g(n,x,y) = \max(x,y,n) - \min(x,y,n)$

Lösung:

$$\mu g(x,y) = \begin{cases} x & \text{falls } x = y, \\ \text{undefiniert}, & \text{sonst.} \end{cases}$$

Aufgabe 5. Seien $h\colon \mathbb{N}^2 \to \mathbb{N}$ und $k\colon \mathbb{N}^3 \to \mathbb{N}$ beliebige primitiv-rekursive Funktionen. Zeigen Sie, dass die folgenden Funktionen ebenfalls primitiv-rekursiv sind.

(a) (2 Punkte) $f\colon \mathbb{N}^2 \to \mathbb{N}$ mit $f(x,y) = 2^{k(x,y,x-y)}$

Lösung: Mit $f'(n) = 2^n$ gilt

$$\begin{aligned} f'(0) &= 1 \\ f'(n+1) &= f'(n) \cdot 2 = mult(f'(n), 2), \end{aligned}$$

daher ist f' primitiv-rekursiv. Durch Einsetzung erhalten wir $f(x,y) = f'(k(x,y,sub(x,y)))$. Also ist f primitiv-rekursiv.

(b) (3 Punkte) $g\colon \mathbb{N}^3 \to \mathbb{N}$ mit $g(x,y,z) = \prod_{i=1}^{x}(i \cdot h(k(z,y,y),i) + (i+yz) \cdot h(y,z))$

Lösung: Es gilt

$$\begin{aligned} g(0,y,z) &= 1 \\ g(x+1,y,z) &= g(x,y,z) \cdot ((x+1) \cdot h(k(z,y,y),x+1) + (x+1+yz) \cdot h(y,z)) \\ &= g(x,y,z) \cdot (s(x) \cdot h(k(z,y,y),s(x)) + (s(x)+yz) \cdot h(y,z)) \end{aligned}$$

daher lässt sich $g(x+1,y,z)$ als Verkettung primitiv-rekursiver Funktionen in den Variablen x,y,z und $g(x,y,z)$ schreiben und somit ist g primitiv-rekursiv.

Beachte: Die Terme innerhalb des Produkts sind unabhängig von x, d.h. nur die Laufvariable i kommt vor. Sonst dürften wir $g(x,y,z)$ nicht einfach rausziehen.

Aufgabe 6. (5 Punkte) Sei $b_n = |bin(n)|$ die Länge der Binärdarstellung von n. Gegeben sei die Funktion $f\colon \mathbb{N} \to \mathbb{N}$ mit

$$f(n) = 2n + 2^{b_n+1}.$$

Geben Sie eine Turingmaschine an, die f berechnet.

Hinweis: Beachten Sie, dass die Eingabezahl n zu Beginn als Binärzahl auf dem Band steht (zum Beispiel steht bei $n = 5$ zu Beginn 101 auf dem Band).

Achten Sie darauf, dass der Leseschreibkopf am Ende auf dem ersten Zeichen der Ausgabe steht.

Lösung: Die Multiplikation mit 2 bedeutet, dass wir an die Binärdarstellung von n eine 0 anhängen müssen. Man beachte, $bin(n)$ hat b_n viele Ziffern, also hat $bin(n)0$ (b_n+1) viele Ziffern. Die Zahl 2^{b_n+1} ist aber eine 1 gefolgt von (b_n+1) 0en, also entspricht die Addition einfach dem Anhängen von 1 vor $bin(n)0$. Mit anderen Worten, die Turingmaschine muss eine 1 vorne und eine 0 hinten anhängen. Der Fall $n = 0$ ist damit ebenfalls abgedeckt.

$M = (\{z_0, z_1, z_f\}, \{0, 1\}, \{0, 1, \square\}, \delta, z_0, \square, \{z_f\})$

$$\begin{aligned}
\delta(z_0, x) &= \{(z_0, x, R)\}, \quad x \in \{0, 1\} \\
\delta(z_0, \square) &= \{(z_1, 0, L)\} \\
\delta(z_1, x) &= \{(z_1, x, L)\}, \quad x \in \{0, 1\} \\
\delta(z_1, \square) &= \{(z_f, 1, N)\}
\end{aligned}$$

Bemerkung: Obige Turingmaschine ist sogar deterministisch. Die Übergänge sind für alle Paare $(Z \backslash E) \times \Gamma$ eindeutig definiert.

Aufgabe 7. (6 Punkte) Betrachten Sie das folgende Problem:

$$L = \{F \mid F \text{ ist eine erfüllbare aussagenlog. Formel der Form } G \rightarrow H\}$$

Zeigen Sie, dass L **NP**-vollständig ist.

Hinweis: Dazu müssen Sie zeigen, dass L in **NP** liegt und dass L **NP**-hart ist. Dafür dürfen Sie nutzen, das die Menge der erfüllbaren aussagenlog. Formeln SAT **NP**-vollständig ist.

Lösung: Sei F' von der Form $G \rightarrow H$. Es gilt $\mathcal{B}(F') = 1$ genau dann wenn $\mathcal{B}(G) = 0$ oder $\mathcal{B}(H) = 1$. Wir definieren also $f(F) = F' = (A \vee \neg A) \rightarrow F$. Ist $F \in SAT$, so ist auch F' erfüllbar (gleiches Modell wie F erweitert um einen beliebigen Wert für A). Falls $F \notin SAT$ ist, so ist auch F' unerfüllbar, denn es gilt dann stets $\mathcal{B}(A \vee \neg A) = 1$ und $\mathcal{B}(F) = 0$ für jede Belegung $\mathcal{B}$. Die Reduktion f ist offenbar in Polynomialzeit berechenbar. Somit haben wir $SAT \leq_p L$ und L ist **NP**-hart.

Es gilt außerdem $L \subseteq SAT$, d.h. wir können überprüfen, ob eine Formel $F \in L$ ist indem wir zunächst testen, ob F erfüllbar ist ($F \in SAT$) und ob F die passende Gestalt hat, also $F = G \rightarrow H$ für aussagenlogische Formeln G und H. Ob F erfüllbar ist geht in **NP** und ob F eine Implikation ist, kann in Linearzeit überprüft werden. Folglich ist $L \in$ **NP**.

Insgesamt ist L also **NP**-vollständig.

Aufgabe 8. (6 Punkte) Zeigen Sie durch geeignetes Anwenden des Markierungsalgorithmus, dass folgende Aussage wahr ist:

$$(\neg A \vee B), (\neg A \vee \neg B \vee C), (\neg B \vee \neg C \vee \neg D \vee E), (\neg B \vee \neg C \vee D) \vDash (E \vee \neg A)$$

Geben Sie an, welche atomaren Formeln in jedem Schritt markiert werden, jeweils mit Begründung.

Lösung: Dies gilt genau dann, wenn

$$(A \to B) \wedge (A \wedge B \to C) \wedge (B \wedge C \wedge D \to E) \wedge (B \wedge C \to D) \wedge (E \to 0) \wedge (1 \to A)$$

unerfüllbar ist.

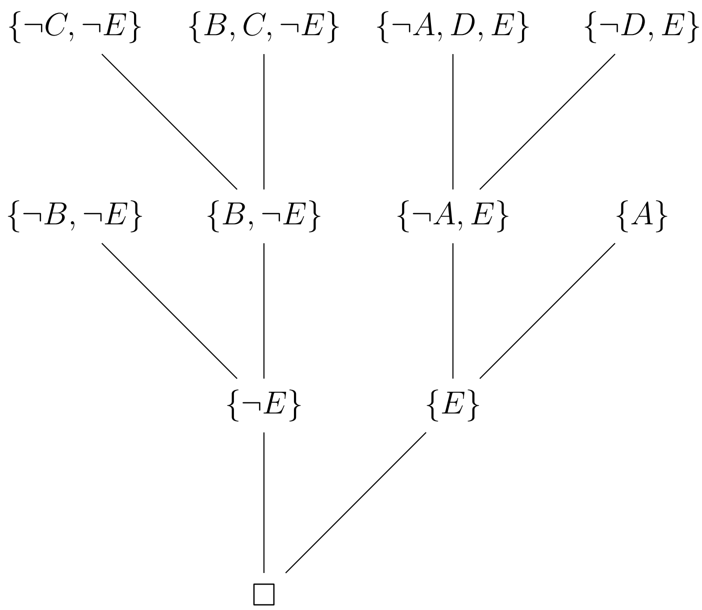
- Markiere A wegen $1 \to A$.
- Markiere B weil A markiert und $A \to B$.
- Markiere C, weil A und B markiert und $A \wedge B \to C$.
- Markiere D, weil B und C markiert und $B \wedge C \to D$.
- Markiere E, weil C und B und D markiert und $B \wedge C \wedge D \to E$.
- Gib unerfüllbar aus, weil E markiert und $E \to 0$.

Daher ist die Aussage wahr.

Aufgabe 9. (6 Punkte) Verwenden Sie das Resolutionsverfahren der Aussagenlogik, um zu zeigen, dass die folgende Klauselmenge unerfüllbar ist:

$$\{\{A\}, \{B, C, \neg E\}, \{\neg A, D, E\}, \{\neg D, E\}, \{\neg C, \neg E\}, \{\neg B, \neg E\}\}$$

Lösung:



Aufgabe 10. Zeigen Sie, dass die folgenden Formeln erfüllbar, aber nicht gültig sind:

(a) (3 Punkte) $F = (\neg\exists x\forall y(\neg P(x,y) \vee P(x,x))) \wedge \forall z(P(f(z),z) \wedge \neg P(z,f(z)))$

Lösung: Sei $U_\mathcal{A} = \mathbb{Z}$, $f^\mathcal{A}(x) = x - 1$, $P^\mathcal{A} = \{(x,y) \mid x < y\}$, dann gilt $\mathcal{A} \models F$, also ist F erfüllbar.
Sei $U_\mathcal{A} = \{0\}$, $f^\mathcal{A}(x,y) = 0$, $P^\mathcal{A} = \emptyset$, dann gilt $\mathcal{A} \not\models F$, also ist F nicht gültig.

(b) (3 Punkte) $G = \exists x\exists y\forall z\neg(R(g(x,x,x)) \vee \neg R(g(y,y,y)) \vee \neg R(g(z,z,a)))$

Lösung: Sei $U_\mathcal{A} = \mathbb{Z}$, $g^\mathcal{A}(x,y,z) = x \cdot y \cdot z$, $R^\mathcal{A} = \{x \mid x \geq 0\}$, $a^\mathcal{A} = 5$, dann gilt $\mathcal{A} \models G$, also ist G erfüllbar.
Sei $U_\mathcal{A} = \{0\}$, $g^\mathcal{A}(x,y,z) = x$, $R^\mathcal{A} = \{0\}$, $a^\mathcal{A} = 0$, dann gilt $\mathcal{A} \not\models G$, also ist G nicht gültig.

Aufgabe 11. (6 Punkte) Gegeben sei die folgende Formel F:

$$\forall x \exists y Q(g(x,y), g(y,x)) \wedge \forall x(\neg R(f(x,z)) \vee R(f(z,x)))$$

Überführen Sie F in eine erfüllbarkeitsäquivalente Aussage in Klauselform.

Lösung: Bereinigen:

$\forall x_1 \exists y_1 Q(g(x_1,y_1), g(y_1,x_1)) \wedge \forall x_2(\neg R(f(x_2,z)) \vee R(f(z,x_2)))$

Freie Variablen ersetzen:

$\forall x_1 \exists y_1 Q(g(x_1,y_1), g(y_1,x_1)) \wedge \forall x_2(\neg R(f(x_2,a_z)) \vee R(f(a_z,x_2)))$

Pränexform:

$\forall x_1 \exists y_1 \forall x_2 (Q(g(x_1,y_1), g(y_1,x_1)) \wedge (\neg R(f(x_2,a_z)) \vee R(f(a_z,x_2))))$

Skolemform:

$\forall x_1 \forall x_2 (Q(g(x_1, f_{y_1}(x_1)), g(f_{y_1}(x_1), x_1)) \wedge (\neg R(f(x_2,a_z)) \vee R(f(a_z,x_2))))$

Klauselform:
$\forall x_1 \forall x_2 (Q(g(x_1, f_{y_1}(x_1)), g(f_{y_1}(x_1), x_1)) \wedge (\neg R(f(x_2,a_z)) \vee R(f(a_z,x_2))))$
(Matrix ist bereits in KNF)

Aufgabe 12. Gegeben sei die Struktur $\mathcal{A} = (U_\mathcal{A}, I_\mathcal{A})$, wobei das Universum gegeben ist durch

$$U_\mathcal{A} = \{\varphi : \mathbb{N} \to \mathbb{N} \mid \varphi \text{ ist eine totale Funktion}\}$$

und die Interpretationsfunktion $I_\mathcal{A}$ definiert ist durch:

- $P^\mathcal{A} = \{\varphi \mid \varphi$ ist WHILE-berechenbar$\}$,
- $Q^\mathcal{A} = \{\varphi \mid \varphi$ ist primitiv rekursiv$\}$,
- $R^\mathcal{A} = \{\varphi \mid \varphi$ ist Turing-berechenbar$\}$,
- $S^\mathcal{A} = \{(\varphi, \psi) \mid \varphi = \mu\psi\}$ (dabei ist μ der Operator der μ-Rekursion),
- $f^\mathcal{A}(\varphi, \psi) = \varphi \circ \psi$ (die Komposition der Funktionen),
- $a^\mathcal{A} = x \mapsto x^2$.

Formulieren Sie die folgenden Aussagen als prädikatenlogische Formeln:

(a) (2 Punkte) Jede primitiv rekursive Funktion ist Turing-berechenbar, aber nicht jede Turing-berechenbare Funktion ist primitiv rekursiv.

Lösung: $\forall x(Q(x) \to R(x)) \land \neg\forall y(R(y) \to Q(y))$

(b) (2 Punkte) Nicht jede Komposition von zwei Turing-berechenbaren Funktionen ist primitiv rekursiv.

Lösung: $\neg\forall x \forall y((R(x) \land R(y)) \to Q(f(x, y))$

(c) (2 Punkte) Die Funktion $x \mapsto x^2$ ist WHILE-berechenbar und die Funktion $x \mapsto x^4$ ist primitiv rekursiv.

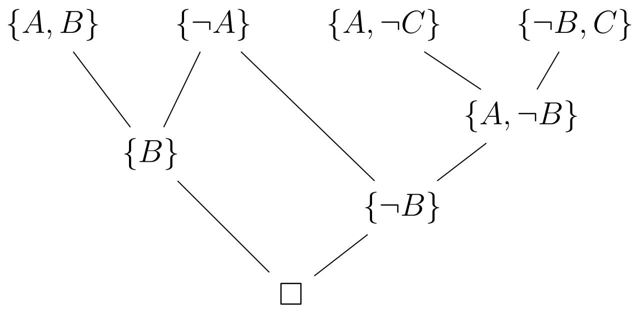
Lösung: $P(a) \land Q(f(a, a))$

(d) (2 Punkte) Es gibt eine totale Funktion φ, für die es eine totale Funktion ψ mit $\mu\varphi = \psi$ gibt.

Lösung: $\exists x \exists y S(x, y)$

Aufgabe 13. (3 Punkte (Bonus))

Gegeben sei die Formel $F = (A \wedge B) \vee \neg A \vee (A \wedge \neg C) \vee (\neg B \wedge C)$. Betrachten Sie den folgenden Resolutionsbeweis:



Die Formel F ist jedoch erfüllbar.

(a) Erklären Sie den scheinbaren Widerspruch.

Lösung: Die Formel ist in DNF, nicht in KNF. Also repräsentieren die Mengen des „Resolutionsbeweises" gar keine Klauseln! Oder ($\vee$) und Und ($\wedge$) müsste man in F vertauschen.

(b) Geben Sie ein Modell für F an.

Lösung: Zum Beispiel definiert $\mathcal{B}(A) = 0$ uns ein Modell für F. Die Werte für B und C sind egal.