6.0 VU Theoretische Informatik 2023W 26.1.2024			
Matrikelnummer	Nachname **Lösung**	Vorname	Gruppe **A**

Aufgabe 1

(a) [**10 Punkte**] Wir definieren das folgende Entscheidungsproblem:

> 2-Halteproblem
> Instanz: Zwei SIMPLE Programme Π_1, Π_2 und ein Input String I.
> Frage: Terminieren beide Programme Π_1 und Π_2 auf dem Input I?

Zeigen Sie, dass dieses Problem semi-entscheidbar ist, indem Sie eine Semi-Entscheidungsprozedur für das 2-Halteproblem skizzieren. Geben Sie auch eine Begründung dafür, dass es sich tatsächlich um eine Semi-Entscheidungsprozedur für dieses Problem handelt, indem Sie kurz das Verhalten der Prozedur auf positiven bzw. negativen Instanzen diskutieren.

Lösung

Semi-Entscheidungsprozedur: Mit Hilfe eines Interpreters für SIMPLE Programme können wir ein Programm Π_i mit folgenden Eigenschaften schreiben:

- Π_i nimmt als Input eine beliebige Instanz des 2-Halteproblems, d.h., den Quellcode von zwei SIMPLE Programmen Π_1, Π_2 und einen Input I für Π_1 und Π_2.
- Π_i simuliert zunächst die Ausführung von Π_1 auf dem Input I.
- Wenn die Simulation von Π_1 auf dem Input I terminiert, dann startet Π_i die Simulation von Π_2 auf dem Input I.
- Wenn die Simulation von Π_2 auf dem Input I terminiert, dann gibt Π_i true aus und terminiert selbst.

Begründung: Wir argumentieren, dass das oben skizzierte Programm Π_i eine Semi-Entscheidungsprozedur für das 2-Halteproblem ist. Dazu unterscheiden wir 2 Fälle:

(1) Angenommen, das Programm Π_i wird mit einer positiven Instanz (Π_1, Π_2, I) des 2-Halteproblems aufgerufen, d.h., beide Programme Π_1, Π_2 terminieren auf dem Input I. Daher wird auch zuerst die Simulation des Programms Π_1 auf dem Input I irgendwann terminieren, und ebenso wird die anschließende Simulation des Programms Π_2 auf dem Input I terminieren. Laut obiger Beschreibung gibt Π_i dann true aus und terminiert selbst. Das ist das korrekte Verhalten für eine Semi-Entscheidungsprozedur auf positiven Instanzen.

(2) Angenommen, das Programm Π_i wird mit einer negativen Instanz (Π_1, Π_2, I) des 2-Halteproblems aufgerufen, d.h., zumindest eines der beiden Programme Π_1 und/oder Π_2 terminiert auf dem Input I nicht. Wir unterscheiden 2 Fälle:

- Falls Π_1 auf dem Input I nicht terminiert, dann wird auch die Simulation des Programms Π_1 auf dem Input I nie terminieren. Das Programm Π_i läuft in diesem Fall endlos.
- Falls Π_1 auf dem Input I terminiert, dann terminiert Π_2 auf dem Input I nicht (laut Annahme terminiert zumindest eines der beiden Programme nicht). In diesem Fall startet Π_i die Simulation von Π_2 auf dem Input I und diese Simulation terminiert nicht. Das Programm Π_i läuft also wiederum endlos.

In beiden Fällen ist das das korrekte Verhalten für eine Semi-Entscheidungsprozedur auf negativen Instanzen.

Bemerkung: Es wären auch andere Formulierungen bzw. Realisierungen der Semi-Entscheidungsprozedur denkbar.

- Zum Beispiel könnte man die oben beschriebene Prozedur auch formal als SIMPLE Programm anschreiben:

  Boolean Π_i (Program Π_1, Program Π_2, String I) {
  call $\Pi_1(I)$;
  call $\Pi_2(I)$;
  return true; }

- Oder man könnte Π_i so definieren, dass beide Programme Π_1 und Π_2 auf dem Input I parallel simuliert werden und Π_i nur hält und true ausgibt, wenn beide Simulationen terminieren.

(b) [2 Punkte] Ist das 2-Halteproblem entscheidbar? Begründen Sie Ihre Antwort kurz (maximal zwei Sätze). Punkte gibt es nur für eine hinreichend begründete Antwort.

Lösung

Nein, das Problem ist unentscheidbar.
Begründung: Das Halteproblem lässt sich auf das 2-Halteproblem reduzieren, indem wir eine beliebige Instanz (Π, I) des Halteproblems auf die Instanz (Π, Π, I) des 2-Halteproblems abbilden.

Bemerkung: Der häufigste Fehler bei dieser Frage war, dass die Unentscheidbarkeit mittels Eigenschaften der Semi-Entscheidungsprozedur von Teil (a) argumentiert wurde, z.B.: "Bei negativen Instanzen läuft das Programm endlos". Aus der Semi-Entscheidbarkeit eines Problems (= positives Resultat) lässt sich nicht seine Unentscheidbarkeit (= negatives Resultat) schließen. Und aus den Eigenschaften eines konkreten Programms kann man nicht auf Eigenschaften aller in Frage kommenden Programme schließen.

Aufgabe 2

(a) [10 Punkte] Sei $L = \{a^k(ba)^n \mid k > n\}$. Vervollständigen Sie den folgenden Beweis, der mit Hilfe des Pumping Lemmas für reguläre Sprachen zeigen soll, dass L nicht regulär ist.

Beweis indirekt. Angenommen, L ist regulär. Sei dann m die Konstante aus dem Pumping Lemma und das gewählte Wort

$w =$ $\boldsymbol{a^{m+1}(ab)^m}$ ______________________________

Dann gilt $w \in L$ und $|w| =$ $\boldsymbol{3m+1}$ ____________________, also $|w| \geq m$.

Wir teilen nun w in xyz so auf, dass $|xy| \leq m$ und $|y| > 0$. Nachdem $|xy| \leq m$, kann xy nur aus

bis zu $\boldsymbol{m}$ Symbolen $\boldsymbol{a}$ ______________________________ bestehen.

Nach dem Pumping Lemma muss aber $xy^iz \in L$ für alle $i \geq 0$ gelten.

Wenn wir nun $i =$ $\mathbf{0}$ ____________________ wählen, müsste – wäre L regulär – auch

$xy^iz =$ $\boldsymbol{a^{m+1-|y|}(ab)^m}$ ______________________________ aus L sein.

Dies ist aber nicht der Fall, da

$\boldsymbol{y}$ aus mindestens einem Symbol $\boldsymbol{a}$ besteht, und damit die Anzahl der Symbole $\boldsymbol{a}$ nicht mehr die Anzahl der (ab)-Blöcke übersteigt. ____________________

____

Wir haben also einen Widerspruch gefunden, L kann somit keine reguläre Sprache sein.

(b) [**2 Punkte**] Ist jede Teilmenge einer regulären Menge selbst wiederum regulär? Begründen Sie Ihre Antwort kurz (maximal zwei Sätze). Punkte gibt es nur für eine hinreichend begründete Antwort.

Lösung

Nein, im Allgemeinen sicher nicht. Ein Gegenbeispiel: die Sprache $L = \{0,1\}^*$ ist regulär, $\{0^n1^n \mid n \geq 0\}$ ist eine Teilmenge von $L = \{0,1\}^*$, aber selbst sicher nicht regulär.

Aufgabe 3

Sei E der **Pr**-Funktionsausdruck $\mathbf{Pr}(C_1^0, S \circ S \circ P_2^2)$.

(a) [**4 Punkte**] Werten Sie E für die Eingabewerte 0, 1, 2 und 3 aus.

Lösung

$E(0) = C_1^0() = 1$
$E(1) = S \circ S \circ P_2^2(0, E(0)) = S \circ S \circ P_2^2(0,1) = S \circ S(1) = 1+1+1 = 3$
$E(2) = S \circ S \circ P_2^2(1, E(1)) = S \circ S \circ P_2^2(1,3) = S \circ S(1) = 3+1+1 = 5$
$E(3) = S \circ S \circ P_2^2(2, E(2)) = S \circ S \circ P_2^2(2,5) = S \circ S(1) = 5+1+1 = 7$

(b) [**2 Punkte**] Welche Funktion wird durch E dargestellt?

Lösung

$E(n) = 2n+1$

(c) [**2 Punkte**] Welche Funktion wird durch μE dargestellt?

Lösung

$\mu E() = \min_{y \geq 0}[E(y) = 0] = \min_{y \geq 0}[2y+1 = 0]$.
μE definiert also die 0-stellige Funktion, deren Wert undefiniert ist.

(d) [2 Punkte] Welche Funktion wird durch $\overline{\mu}E$ dargestellt?

Lösung

$\overline{\mu}E(n) = \min_{y\geq 0}[E(y) = n] = \min_{y\geq 0}[2y + 1 = n]$.
Es gilt daher

$$\overline{\mu}E(n) = \begin{cases} \frac{n-1}{2} & \text{falls } n \text{ ungerade} \\ \text{undefiniert} & \text{sonst} \end{cases}$$

(e) [2 Punkte] Ist es entscheidbar, ob eine gegebene Turingmaschine M die durch E dargestellte Funktion berechnet? Begründen Sie Ihre Antwort kurz. Punkte gibt es nur für eine hinreichend begründete Antwort.

Lösung

Die Menge $\{E\}$ ist eine (extensionale) nicht-triviale Funktions-Eigenschaft. Daher ist der Satz von Rice für Funktionen anwendbar. Das Problem ist folglich unentscheidbar.

Aufgabe 4 [12 Punkte]

Ein ungerichteter Graph $G = (V, E)$ heißt *Spiegelgraph*, wenn folgendes gilt:

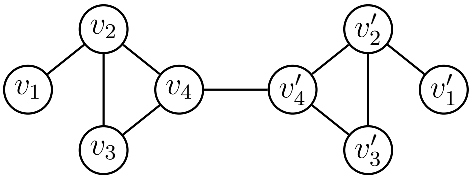
(1) $V = V_1 \cup V_2$ mit $|V_1| = |V_2|$, $V_1 \cap V_2 = \emptyset$,

(2) $E = E_1 \cup E_2 \cup \{[x, y]\}$ mit $E_i \subseteq V_i \times V_i$ $(i \in \{1, 2\})$, $x \in V_1$, $y \in V_2$.

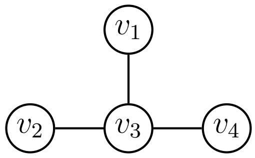
(3) es gibt eine Bijektion $m: V_1 \to V_2$, sodass $[v_i, v_j] \in E_1$ genau dann, wenn $[m(v_i), m(v_j))] \in E_2$. Für die Knoten x, y gilt $m(x) = y$.

In anderen Worten, ein Spiegelgraph besteht aus zwei gespiegelten „Kopien" ein und desselben Graphen, welche durch eine „Brückenkante" (x, y) verbunden sind.

Beispiele:



G_1



G_2

G_1 ein Spiegelgraph mit $m(v_i) = v'_i$ für $1 \leq i \leq 4$ und Brückenkante (v_4, v'_4).
G_2 hingegen ist kein Spiegelgraph, da keine Kante als Brückenkante fungieren kann.

Wir betrachten das klassische Dreifärbbarkeitsproblem (3-Col) und die auf Spiegelgraphen eingeschränkte Variante davon (3-Col-S).

3-Col Instanz: Ein ungerichteter Graph $G = (V, E)$. Frage: Gibt es eine Funktion $f: V \to \{0, 1, 2\}$, sodass $f(v_i) \neq f(v_j)$ für alle Kanten $[v_i, v_j] \in E$ gilt?

3-Col-S Instanz: Ein Spiegelgraph $G = (V, E)$. Frage: Gibt es eine Funktion $f: V \to \{0, 1, 2\}$, sodass $f(v_i) \neq f(v_j)$ für alle Kanten $[v_i, v_j] \in E$?

Betrachten Sie die folgende Reduktion R von 3-COL nach 3-COL-S, die jedem Graphen $G = (V, E)$ einen Spiegelgraph $R(G) = (V^*, E^*)$ mit

$$\begin{aligned} V^* &= V \cup \{v' \mid v \in V\} \\ E^* &= E \cup \{(v_i', v_j') \mid (v_i, v_j) \in E\} \cup \{(v_b, v_b')\}, \end{aligned}$$

zuordnet, wobei $v_b \in V$ ein beliebiger Knoten in G ist.

Zeigen Sie die Korrektheit der Reduktion, also:

G ist eine positive Instanz von 3-COL $\iff$ $R(G)$ ist eine positive Instanz von 3-COL-S.

Lösung

$\Longrightarrow$: Sei G eine positive Instanz von 3-COL. Dann gibt es eine Funktion $f\colon V \to \{0, 1, 2\}$, so dass für alle Kanten $[v_i, v_j] \in E$ gilt: $f(v_i) \neq f(v_j)$. Wir definieren $f^* : V^* \to \{0, 1, 2\}$ mit $f^*(v) = f(v)$ und $f^*(v') = f(v) + 1 \mathsf{\ mod\ } 3$ für alle $v \in V$.
Wir zeigen, dass f^* eine gültige Färbung für $R(G)$ ist. Sei also $[u, w]$ eine Kante aus E^*. Gemäß der Definition von E^* unterscheiden wir folgende Fälle: (1) $u, w \in V$; (2) $u = v_i'$ und $w = v_j'$ für $v_i, v_j \in V$; (3) $u = v_b$, $w = v_b'$.
Für den Fall (1) gilt laut Definition von f^* und aufgrund der Annahme, dass f eine gültige Färbung ist: $f^*(u) = f(u) \neq f(w) = f^*(w)$. Für den Fall (2) gilt laut Definition von E^*, $[v_i, v_j] \in E$. Laut Annahme haben wir $f(v_i) \neq f(v_j)$. Es folgt klarerweise $f(v_i) + 1 \mathsf{\ mod\ } 3 \neq f(v_j) + 1 \mathsf{\ mod\ } 3$ und daher gemäß der Definition von f^*, dass $f^*(u) \neq f^*(w)$. Für den Fall (3) gilt laut Definition von f^*, dass $f^*(w) = f^*(u) + 1 \mathsf{\ mod\ } 3$, und daher, dass $f^*(w) \neq f^*(u)$.
Wir haben somit gezeigt, dass für alle Kanten $[u, w] \in E^*$, $f^*(u) \neq f^*(w)$ gilt. f^* ist daher eine gültige Färbung für $R(G)$; und $R(G)$ daher eine positive Instanz von 3-COL-S.
$\Longleftarrow$: Sei $R(G)$ eine positive Instanz von 3-COL-S. Da G ein Subgraph von $R(G)$, ist klarerweise G auch eine positive Instanz von 3-COL.

Aufgabe 5 [12 Punkte]

Zeigen Sie mit Hilfe der Annotierungsregeln, dass die folgende Korrektheitsaussage wahr hinsichtlich totaler Korrektheit ist. Verwenden Sie $0 \leq j \leq n \wedge k = 2j + 1 \wedge q = j^2$ als Invariante und $n - j$ als Variante.

Eine nützliche Annotierungsregel:
`while` e `do` $\Pi \quad \mapsto \quad \{Inv\}$ `while` e `do` $\{Inv \wedge e \wedge t = t_0\}\ \Pi\ \{Inv \wedge (e \supset 0 \leq t < t_0)\}\ \{Inv \wedge \neg e\}$

$\{\, \boldsymbol{n \geq 0} \,\}$
{{{
$q := 0$;
$k := 1$};
$j := 0$};
`while` $j \neq n$ `do` {{
    $q := q + k$;
    $k := k + 2$};
    $j := j + 1$
}}
$\{\, \boldsymbol{q = n^2} \,\}$

Lösung

Sei $Inv = 0 \le j \le n \land k = 2j+1 \land q = j^2$ und $t = n - j$.

$\{\, Pre\colon n \ge 0 \,\}$
$\{\, F_4\colon Inv\begin{bmatrix}0\\j\end{bmatrix}\begin{bmatrix}1\\k\end{bmatrix}\begin{bmatrix}0\\q\end{bmatrix} \,\}$
`{{{`
$q \coloneqq 0$;
$\{\, Inv\begin{bmatrix}0\\j\end{bmatrix}\begin{bmatrix}1\\k\end{bmatrix} \,\}$
$k \coloneqq 1$`};`
$\{\, Inv\begin{bmatrix}0\\j\end{bmatrix} \,\}$
$j \coloneqq 0$`};`
$\{\, Inv \,\}$
`while` $j \ne n$ `do {{`
$\quad\{\, F_1\colon Inv \land j \ne n \land t = t_0 \,\}$
$\quad\{\, F_3\colon (Inv \land (j \ne n \supset 0 \le t < t_0))\begin{bmatrix}j+1\\j\end{bmatrix}\begin{bmatrix}k+2\\k\end{bmatrix}\begin{bmatrix}q+k\\q\end{bmatrix} \,\}$
$\quad q \coloneqq q + k$;
$\quad\{\, (Inv \land (j \ne n \supset 0 \le t < t_0))\begin{bmatrix}j+1\\j\end{bmatrix}\begin{bmatrix}k+2\\k\end{bmatrix} \,\}$
$\quad k \coloneqq k + 2$`};`
$\quad\{\, (Inv \land (j \ne n \supset 0 \le t < t_0))\begin{bmatrix}j+1\\j\end{bmatrix} \,\}$
$\quad j \coloneqq j + 1$
$\quad\{\, Inv \land (j \ne n \supset 0 \le t < t_0) \,\}$
`}}`
$\{\, F_2\colon Inv \land \neg j \ne n \,\}$
$\{\, Post\colon q = n^2 \,\}$

Wir müssen die Gültigkeit von drei Implikationen nachweisen.

$\boldsymbol{Pre \supset F_4}$: $n \ge 0 \supset Inv\begin{bmatrix}0\\j\end{bmatrix}\begin{bmatrix}1\\k\end{bmatrix}\begin{bmatrix}0\\q\end{bmatrix}$

$$n \ge 0 \supset (0 \le j \le n \land k = 2j+1 \land q = j^2)\begin{bmatrix}0\\j\end{bmatrix}\begin{bmatrix}1\\k\end{bmatrix}\begin{bmatrix}0\\q\end{bmatrix}$$
$$n \ge 0 \supset 0 \le 0 \le n \land 1 = 2\cdot 0 + 1 \land 0 = 0^2$$

Die letzte Formel ist offenbar gültig: $0 \le 0$, $1 = 2\cdot 0 + 1$ und $0 = 0^2$ sind immer wahr, während $0 \le n$ wegen der Prämisse $n \ge 0$ gilt.

$\boldsymbol{F_1 \supset F_3}$:

$$Inv \land j \ne n \land t = t_0 \supset (Inv \land (j \ne n \supset 0 \le t < t_0))\begin{bmatrix}j+1\\j\end{bmatrix}\begin{bmatrix}k+2\\k\end{bmatrix}\begin{bmatrix}q+k\\q\end{bmatrix}$$

Wir zerlegen diese Implikation in zwei Teile:

$$Inv \land j \ne n \supset Inv\begin{bmatrix}j+1\\j\end{bmatrix}\begin{bmatrix}k+2\\k\end{bmatrix}\begin{bmatrix}q+k\\q\end{bmatrix} \quad \text{(partielle Korrektheit)}$$
$$Inv \land j \ne n \land t = t_0 \supset (j \ne n \supset 0 \le t < t_0)\begin{bmatrix}j+1\\j\end{bmatrix}\begin{bmatrix}k+2\\k\end{bmatrix}\begin{bmatrix}q+k\\q\end{bmatrix} \quad \text{(Termination)}$$

Für die Implikation betreffend partielle Korrektheit erhalten wir:

$$Inv \land j \ne n \supset 0 \le j{+}1 \le n \land k{+}2 = 2(j{+}1) + 1 \land q{+}k = (j{+}1)^2$$

Wir zeigen, dass jedes Konjunkt auf der rechte Seite der Implikation wahr ist.

$0 \leq j{+}1$	folgt aus $0 \leq j$ auf der linken Seite (Teil von *Inv*)
$j+1 \leq n$	folgt aus $j \leq n$ (Teil von *Inv*) und $j \neq n$
$k{+}2 = 2(j{+}1)+1$	folgt aus $k = 2j+1$ (Teil von *Inv*), Addition von 2 und Umformung
$q{+}k = (j{+}1)^2$	folgt aus $q = j^2$ und $k = 2j+1$ Teil von *Inv*) durch Addition

Für die Implikation betreffend Termination erhalten wir:

$$Inv \land j \neq n \land n - j = t_0 \supset (j{+}1 \neq n \supset 0 \leq n-(j{+}1) < t_0)$$
$$Inv \land j \neq n \land n - j = t_0 \land j{+}1 \neq n \supset 0 \leq n-(j{+}1) < t_0$$

Wir zeigen, dass jedes Konjunkt auf der rechte Seite der Implikation wahr ist.

$0 \leq n-(j{+}1)$	äquivalent zu $j < n$, folgt aus $j \leq n$ (in *Inv*) und $j \neq n$
$n-(j{+}1) < t_0$	ist wegen $n - j = t_0$ äquivalent zu $j < j{+}1$ und daher gültig

$\boldsymbol{F_2} \supset \boldsymbol{Post}$: $Inv \land \neg j \neq n \supset q = n^2$
$\neg j \neq n$ ist gleichbedeutend mit $j = n$, was zusammen mit $q = j^2$ (in *Inv*) die Formel $q = n^2$ ergibt.

Anmerkung: Diese Musterlösung ist ausführlicher, als eine korrekte Lösung bei der Prüfung sein müsste. Etwa ist es nicht notwendig, zuerst die beabsichtigten Ersetzungen anzuschreiben und diese dann erst später durchzuführen. Auch können graphische Hilfsmittel (eingekreiste Formelteile, Pfeile) verwendet werden, um anzuzeigen, aus welchen Formelteile welche anderen folgen.