RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

WS 2012/13
Blatt 1

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte)

Das Partition-Problem besteht darin, zu entscheiden, ob eine gegebene Menge von natürlichen Zahlen so in zwei Teile partitioniert werden kann, dass die Summen über die jeweiligen Elemente der einzelnen Teile gleich groß sind. Die Sprache $L_{\text{Partition}}$ enthalte genau jene Zahlmengen, für die die genannte Eigenschaft gilt.
Geben Sie eine formale Darstellung für die Sprache $L_{\text{Partition}}$ an. Machen Sie sich dabei insbesondere Gedanken zur Kodierung der Eingabe, zur Eingabelänge und zum Eingabealphabet.

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte)

Ein Hamiltonpfad in einem Graphen G ist ein Pfad in G, der jeden Knoten genau einmal besucht. Die Sprache des Hamiltonpfad-Problems $L_{\text{Hamiltonpfad}}$ enthalte alle Graphen, die mindestens einen Hamiltonpfad besitzen.
Geben Sie eine formale Darstellung für die Sprache $L_{\text{Hamiltonpfad}}$ an. Machen Sie sich dabei insbesondere Gedanken zur Kodierung der Eingabe, zur Eingabelänge und zum Eingabealphabet.

RWTH Aachen **WS 2012/13**
Lehrgebiet Theoretische Informatik Blatt 1
Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte)

Eine Clique ist eine Menge von Knoten in einem Graphen die alle paarweise durch eine Kante verbunden sind. Darauf aufbauend ist eine k-Clique eine Clique bestehend aus k Knoten.
Die Sprache des k-Clique Entscheidungsproblems $L_{\text{k-Clique}}$ enthalte alle Graphen, die eine k-Clique enthalten.
Geben Sie eine formale Darstellung für die Sprache $L_{\text{k-Clique}}$ an. Machen Sie sich dabei insbesondere Gedanken zur Kodierung der Eingabe, zur Eingabelänge und zum Eingabealphabet.

RWTH Aachen **WS 2012/13**
Lehrgebiet Theoretische Informatik Blatt 1
Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte)

Ein Hamiltonkres in einem Graphen G ist ein Kreis in G, der jeden Knoten genau einmal besucht. Die Sprache des Hamiltonkreis-Problems L_{Hamilton} enthalte alle Graphen, die mindestens einen Hamiltonkreis besitzen.
Geben Sie eine formale Darstellung für die Sprache $L_{\text{Hamiltonkreis}}$ an. Machen Sie sich dabei insbesondere Gedanken zur Kodierung der Eingabe, zur Eingabelänge und zum Eingabealphabet

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte)

Das Subset Sum Entscheidungsproblem besteht darin, zu entscheiden, ob aus einer gegebene Menge natürlicher Zahlen eine Teilmenge ausgewählt werden kann, sodass die Summe über die Teilmenge k ist. Die Sprache $L_{\text{Subset Sum}}$ enthalte genau jene Zahlmengen, für die die genannte Eigenschaft gilt.
Geben Sie eine formale Darstellung für die Sprache $L_{\text{Subset Sum}}$ an. Machen Sie sich dabei insbesondere Gedanken zur Kodierung der Eingabe, zur Eingabelänge und zum Eingabealphabet.

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte)

Geben sie zu der folgenden TM M an, welche Ausgabe auf gegebenem Wort w berechnet wird.

$$M = (\{q_0, q_1, \bar{q}\}, \{0, 1\}, \{0, 1, B\}, B, q_0, \bar{q}, \delta)$$

δ	0	1	B
q_0	$(q_0, 0, R)$	$(q_0, 1, R)$	(q_1, B, L)
q_1	$(\bar{q}, 0, R)$	$(q_1, 1, L)$	(q_0, B, R)

$$w = 010111$$

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte)

Geben sie zu der folgenden TM M an, welche Konfigurationen auf gegebener Eingabe w erreicht werden.

$$M = (\{q_0, q_1, \bar{q}\}, \{0, 1\}, \{0, 1, B\}, B, q_0, \bar{q}, \delta)$$

δ	0	1	B
q_0	$(q_0, 0, R)$	$(q_0, 1, R)$	(q_1, B, L)
q_1	$(\bar{q}, 0, R)$	$(q_1, 1, L)$	(q_0, B, R)

$$w = 0$$

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte)

Geben sie eine TM M an, die nur auf der leeren Eingabe ϵ hält.

Achten sie insbesondere auf formale Korrektheit der Lösung. Beschreiben sie außerdem kurz die Funktionsweise ihrer TM.

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

WS 2012/13
Blatt 2

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte)
Geben sie eine TM M an, die die Funtkion $\{0,1\}^* \to \{0,1\}^*$, $w \to w1$ berechnet.
Achten sie insbesondere auf formale Korrektheit der Lösung. Beschreiben sie außerdem kurz die Funktionsweise ihrer TM.

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte) Zeigen oder widerlegen sie, dass die folgende Sprache entscheidbar ist.

$$A = \{\langle M\rangle w \mid M \text{ akzeptiert } w\}$$

Verwenden Sie *nicht* den Satz von Rice.

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

WS 2012/13
Blatt 3

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte) Zeigen oder widerlegen sie, dass die folgende Sprache entscheidbar ist.

$$A_L = \{\, \langle M \rangle \mid \text{ auf der leeren Eingabe bewegt } M \text{ den Kopf nie nach links} \,\}$$

Verwenden Sie *nicht* den Satz von Rice.

RWTH Aachen **WS 2012/13**
Lehrgebiet Theoretische Informatik Blatt 3
Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte) Zeigen oder widerlegen sie, dass die folgende Sprache entscheidbar ist.

$$A_B = \{\, \langle M \rangle \mid \text{auf der leeren Eingabe schreibt } M \text{ nur Blanks auf das Band} \,\}$$

Verwenden Sie *nicht* den Satz von Rice.

RWTH Aachen **WS 2012/13**
Lehrgebiet Theoretische Informatik Blatt 3
Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte) Gegeben ist das unten stehende RAM-Programm. Beschreibe kurz welche Funktion bei Eingabe einer natürlichen Zahl n berechnet wird.

```
  CLOAD 0
  STORE 2 // i=0
  CLOAD 1
  STORE 3 // j=1
$while
  LOAD 1
  SUB 3  // c(0)=n-j
  IF c(0) < 0 GOTO end
  CLOAD 1
  ADD 2
  STORE 2 // i=i+1
  LOAD 3
  CMULT 2
  STORE 3 // j=j*2
  GOTO while
$end
  LOAD 2
  CSUB 1
  STORE 1
  END
```

Anstatt die absoluten Sprungadressen anzugeben, verwenden wir zur Vereinfachung und besseren Lesbarkeit die von Assemblersprachen bekannten Sprungmarken.

RWTH Aachen **WS 2012/13**
Lehrgebiet Theoretische Informatik Blatt 3
Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte)
Sei
$$A = \{\langle M \rangle w \mid M \text{ akzeptiert } w\}.$$
Zeigen Sie mit Hilfe der Unterprogrammtechnik, dass A nicht entscheidbar ist.

RWTH Aachen **WS 2012/13**
Lehrgebiet Theoretische Informatik Blatt 4
Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte) Zeigen oder widerlegen sie, dass die folgende Sprache entscheidbar ist. $H_{\leq 42} = \{\langle M \rangle \mid M \text{ hält nach höchstens 42 Schritten auf jeder Eingabe}\}$

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte) Zeigen oder widerlegen sie, dass die folgende Sprache entscheidbar ist.
$\mathrm{TAPE}_{\mathrm{positiv}} = \{\langle M\rangle\, w \mid M \text{ nutzt bei Eingabe } w \text{ nur Bandzellen mit Index } i \geq 0\}$

Lösungsvorschlag $\mathrm{TAPE}_{\mathrm{positiv}}$ ist nicht entscheidbar. Jede TM M kann durch eine TM M' ersetzt werden, die ein einseitig beschränktes Arbeitsband benutzt. M' wiederum kann zur TM M'' erweitert werden, die genau dann, wenn M' hält, nach links bis zur Bandposition -1 läuft. Sei H das Halteproblem. Damit gilt: $\langle M''\rangle w \in \mathrm{TAPE}_{\mathrm{positiv}}$ genau dann, wenn $\langle M\rangle w \notin H$. Ein Algorithmus für $\mathrm{TAPE}_{\mathrm{positiv}}$ würde daher auch das Halteproblem entscheiden (für Unterprogrammtechnik muss zu Beginn nur noch ein einfacher Syntaxcheck durchgeführt werden).

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte) Zeigen oder widerlegen sie, dass die folgende Sprache entscheidbar ist.

$$D_H = \{w \in \{0,1\}^* \mid w = w_i \text{ und } M_i \textit{ hält} \text{ nicht auf } w_i\}.$$

Lösungsvorschlag Sei M_{D_H} die Turing-Maschine, die D_H entscheidet. Sei $M_j = M'_{D_H}$ die Turing-Maschine, die sich genauso verhält wie M_{D_H}, aber in eine Endlosschleife fällt, falls M_{D_H} nicht akzeptiert.

- Fall 1: Sei $w_j \in D_H$. Es folgt aus der Definition von D_H, dass M_j nicht auf w_j hält. Aus der Definition von M_j folgt dann, dass w_j von M_{D_H} nicht akzeptiert wird, dass also $w_j \notin D_H$ ist.
- Fall 2: Sei $w_j \notin D_H$. Es folgt aus der Definition von D_H, dass M_j auf w_j hält. Aus der Definition von M_j folgt, dass M_{D_H} auf w_j akzeptiert. Aus der Definition von M_{D_H} folgt dann, dass $w_j \in D_H$ ist.

Somit ist durch Widerspruch gezeigt, dass es kein solches M_{D_H} geben kann, und damit D_H nicht entscheidbar ist.

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte) Zeigen oder widerlegen sie, dass die folgende Sprache entscheidbar ist. $L_{q_0} = \{\langle M\rangle \mid M \text{ verlässt Zustand } q_0 \text{ bei leerer Eingabe}\}$

Lösungsvorschlag L_{q_0} ist rekursiv. Durch Betrachten der Zustandsübergangsfunktion kann man für eine gegebene Turingmaschine M wie folgt entscheiden, ob M in L_{q_0} enthalten ist.
Gilt $\delta_M(q_0, B) = (q_i, X, D)$ für ein $i \neq 0, X \in \Gamma, D \in \{L, N, R\}$, so ist offensichtlich $\langle M\rangle \in L_{q_0}$. Bleibt M jedoch in Zustand q_0, so kann sich der Kopf nicht von seiner Position bewegen, ohne dass M in eine Endlosschleife läuft. Wie in Aufgabe 2.4(b) wissen wir nun also aufgrund dieser Platzbeschränkung, dass M nach spätestens $|\Gamma|$ vielen Schritten den Zustand wechselt oder endlos läuft. Also können wir M auf ϵ simulieren.

RWTH Aachen **WS 2012/13**
Lehrgebiet Theoretische Informatik Blatt 4
Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte) Gegeben ist das unten stehende RAM-Programm. Beschreibe kurz welche Funktion bei Eingabe einer natürlichen Zahl n berechnet wird.

```
  CLOAD 0
  STORE 2 // i=0
  CLOAD 1
  STORE 3 // j=1
$while
  LOAD 1
  SUB 3  // c(0)=n-j
  IF c(0) < 0 GOTO end
  CLOAD 1
  ADD 2
  STORE 2 // i=i+1
  LOAD 3
  CMULT 2
  STORE 3 // j=j*2
  GOTO while
$end
  LOAD 2
  CSUB 1
  STORE 1
  END
```

Anstatt die absoluten Sprungadressen anzugeben, verwenden wir zur Vereinfachung und besseren Lesbarkeit die von Assemblersprachen bekannten Sprungmarken.

Lösungsvorschlag Es wird
$lfloor \log(n)\rfloor$ berechnet.

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

WS 2012/13
Blatt 5

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte)
Wie lautet der Satz von Rice?

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

WS 2012/13
Blatt 5

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)

Geben Sie eine Sprache an, die rekursiv, aber nicht rekursiv aufzählbar ist (ohne Beweis).

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

WS 2012/13
Blatt 5

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte)

Zeigen Sie, daß das spezielle Halteproblem

$$H_\epsilon = \{\langle M\rangle \mid M \text{ hält auf der leeren Eingabe}\}$$

rekursiv aufzählbar ist.

RWTH Aachen **WS 2012/13**
Lehrgebiet Theoretische Informatik Blatt 5
Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte)
Es seien L_1 und L_2 zwei rekursive Sprachen.
Beweisen Sie, daß dann auch $L_1 \cap L_2$ rekursiv ist.

Übung zur Vorlesung Berechenbarkeit und Komplexität

Alle Antworten sind zu beweisen!

Aufgabe 1 (10 Punkte)
Es seien L_1 und L_2 zwei rekursiv aufzählbare Sprachen.
Beweisen Sie, daß dann auch $L_1 \cap L_2$ rekursiv aufzählbar ist.

Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)
Welche der folgenden Sprachen sind rekursiv und welche sind rekursiv aufzählbar?
Sie brauchen diesmal keinen Beweis anzugeben.

Wir definieren wie immer $L(M) = \{\, w \mid M \text{ akzeptiert } w \,\}$.

1. $H_\epsilon = \{\, \langle M \rangle \mid M \text{ hält auf der leeren Eingabe} \,\}$
2. $D = \{\, w \mid \text{es gibt eine Zahl } i \text{ mit } w_i = w,\ M_i = M \text{ und } w_i \in L(M_i) \,\}$
3. das Postsche Korrespondenzproblem
4. $\{(G_1, G_2) \mid G_1 \text{ und } G_2 \text{ sind kontextfreie Grammatiken und } L(G_1) \cup L(G_2) = \emptyset \,\}$
5. $\{\, \langle M \rangle \mid L(M) \text{ endlich} \,\}$
6. $\{\, \langle M \rangle \mid L(M) \text{ ist rekursiv aufzählbar} \,\}$

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

WS 2012/13
Blatt 6

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)
Welche der folgenden Sprachen sind rekursiv und welche sind rekursiv aufzählbar?
Sie brauchen diesmal keinen Beweis anzugeben.

Wir definieren wie immer $L(M) = \{\, w \mid M \text{ akzeptiert } w \,\}$.

1. $U = \{\, \langle M \rangle w \mid w \in L(M) \,\}$
2. $\{(G_1, G_2) \mid G_1 \text{ und } G_2 \text{ sind kontextfreie Grammatiken und } L(G_1) \cap L(G_2) \neq \emptyset \,\}$
3. $\{(G_1, G_2) \mid G_1 \text{ und } G_2 \text{ sind kontextfreie Grammatiken und } L(G_1) \cup L(G_2) = \emptyset \,\}$
4. die Frage, ob eine diophantische Gleichung eine Lösung hat
5. $\{\, \langle M \rangle \mid L(M) \text{ endlich} \,\}$
6. $\{\, \langle M \rangle \mid L(M) = \emptyset \,\}$

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

WS 2012/13
Blatt 6

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)

Welche der folgenden Sprachen sind rekursiv und welche sind rekursiv aufzählbar?
Sie brauchen diesmal keinen Beweis anzugeben.

Wir definieren wie immer $L(M) = \{\, w \mid M \text{ akzeptiert } w \,\}$.

1. $H_{all} = \{\, \langle M \rangle \mid M \text{ hält auf jeder Eingabe} \,\}$
2. $D = \{\, w \mid \text{es gibt eine Zahl } i \text{ mit } w_i = w,\ M_i = M \text{ und } w_i \in L(M_i) \,\}$
3. $\{(G_1, G_2) \mid G_1 \text{ und } G_2 \text{ sind kontextfreie Grammatiken und } L(G_1) \cup L(G_2) = \emptyset \,\}$
4. die Frage, ob eine diophantische Gleichung eine Lösung hat
5. $\{\, \langle M \rangle \mid M \text{ hat mehr als 1000 Zustände} \,\}$
6. $\{\, \langle M \rangle \mid L(M) = \emptyset \,\}$

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

WS 2012/13
Blatt 6

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)

Definieren Sie das Postsche Korrespondenzproblem.

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

WS 2012/13
Blatt 6

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)
Beweisen Sie, daß das Postsche Korrespondenzproblem rekursiv aufzählbar ist.

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

WS 2012/13
Blatt 6

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)

Geben Sie die Syntax der Sprache WHILE-Programmiersprache an.

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)
Gegeben eine binär kodierte Eingabezahl x welche Funktion $f_M(x)$ berechnet folgende TM?

$$M = (\{q_0, q_1, q_2, \bar{q}\}, \{0,1\}, \{0,1,B\}, B, q_0, \bar{q}, \delta)$$

δ	0	1	B
q_0	$(q_0, 0, R)$	$(q_0, 1, R)$	(q_1, B, L)
q_1	(q_2, B, L)	(q_2, B, L)	$(q_0, 0, N)$
q_2	$(q_2, 0, L)$	$(q_2, 1, L)$	$(\bar{q}, B, R)$

Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)

Geben Sie an welche Funktion das folgende WHILE-Programm berechnet.

Eingabe: $x_1 \in \mathbb{N}$

$x_2 := x_1 + 0;$
WHILE $x_2 \neq 0$ DO
  $x_1 := x_1 + 1;$
  $x_2 := x_2 - 1$
END;
$x_0 := x_1 + 1;$

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)
Geben Sie an welche Funktion das folgende WHILE-Programm berechnet.

Eingabe: $x_1 \in \mathbb{N}$

```
x2 := x1 + 0;
WHILE x2 ≠ 0 DO
    x3 := x1 + 0;
    WHILE x3 ≠ 0 DO
        x4 := x4 + 1;
        x3 := x3 − 1
    END;
    x1 := x2 − 1
END;
x0 := x4 + 0;
```

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)

Welche Sprache wird von der folgenden TM entschieden?

$$M = (\{q_0, q_1, \bar{q}\}, \{0, 1\}, \{0, 1, B\}, B, q_0, \bar{q}, \delta)$$

δ	0	1	B
q_0	$(q_1, 0, R)$	$(q_0, 1, R)$	$(\bar{q}, 1, N)$
q_1	$(q_0, 0, R)$	$(\bar{q}, 0, N)$	$(\bar{q}, 0, N)$

Hinweis: Die Sprache kann auch als regulärer Ausdruck angegeben werden.

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)
Zeigen oder widerlegen Sie, dass die folgende Sprache entscheidbar ist.

$\mathrm{TAPE}_{\mathrm{positiv}} = \{\langle M\rangle\, w \mid M \text{ besucht keine Bandzellen links von der Eingabe}\}$

RWTH Aachen **WS 2012/13**
Lehrgebiet Theoretische Informatik Blatt 8
Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)

Es seien L_1 und L_2 zwei rekursiv aufzählbare Sprachen.
Beweisen Sie, daß dann auch $L_1 \cap L_2$ rekursiv aufzählbar ist.

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

WS 2012/13
Blatt 8

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)

Gibt es eine Sprache, die rekursiv, aber nicht rekursiv aufzählbar ist? (ohne Beweis).

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

WS 2012/13
Blatt 8

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)
Zeigen Sie, daß das spezielle Halteproblem

$$H_\epsilon = \{\langle M\rangle \mid M \text{ hält auf der leeren Eingabe}\}$$

rekursiv aufzählbar ist.

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

WS 2012/13
Blatt 8

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)

Geben Sie eine TM an, die nicht hält und bei der sich keine Konfiguration wiederholt unabhängig von der Eingabe.

Achten Sie insbesondere auf formale Korrektheit der Lösung. Beschreiben Sie außerdem kurz die Funktionsweise ihrer TM.

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

WS 2012/13
Blatt 9

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)
Geben Sie zum folgenden Entscheidungsproblem ein Zertifikat und einen Polynomialzeitverifizierer an. Beschreiben Sie dazu im Detail die Kodierung und die Länge des Zertifikates und die Arbeitsweise des Verifizierers.

VERTEXCOVER
Eingabe: Ein Graph $G = (V, E)$ und eine Zahl $b \in \mathbb{N}$.
Ausgabe: Ja, gdw. es eine Knotenmenge $K \subseteq V$ mit $\lvert K\rvert \leq b$ gibt, so dass jede Kante in E durch mind. einen Knoten abgedeckt ist.

Lösungsvorschlag.
Zertifikat und Länge: Das Zertifikat ist eine Teilmenge V' der Knoten von G, die ein Vertex Cover der Größe höchstens k bilden. Es sei $V = \{v_1, \ldots, v_n\}$. Die Kodierung des Zertifikats ist ein Bitstring $B = (b_1, \ldots, b_n) \in \{0, 1\}^n$ der Länge $|V| = n$, wobei $b_i = 1 \Leftrightarrow v_i \in V'$ gilt.
Polynomialzeitverifizierer: Der Zertifizierer muss testen, ob das Zertifikat die richtige Form hat und ob es ein gültiges Vertex Cover kodiert.

(1) Gilt $B \in \{0, 1\}^n$?

(2) Für alle $i \in \{1, \ldots, n\}$ wird getestet, ob für alle Kanten der Form $(v_i, v_j) \in E$ der Knoten v_j in V' ist, also ob $v_j \in V'$ gilt.

Laufzeit: In Schritt 1 wird die gesamte Eingabe überprüft, was in Zeit $\mathcal{O}(n)$ erledigt wird. In Schritt 2 werden von jedem Knoten die Nachbarn untersucht. Ein Knoten kann maximal $n - 1$ viele Nachbarn haben, jeder dieser Nachbarn kann durch Betrachten der Adjazenzmatrix identifiziert werden. Diese Identifikation läuft in Zeit $\mathcal{O}(n^2)$ ab, bei maximal $n - 1$ vielen Nachbarn ergibt sich damit ein Aufwand von $\mathcal{O}(n^3)$. Da auf diese Art alle Knoten bearbeitet werden, kann Schritt 2 in Zeit $\mathcal{O}(n^4)$ abgearbeitet werden. Der Verifizierer läuft folglich in Polynomialzeit.

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte) Geben Sie zum folgenden Entscheidungsproblem ein Zertifikat und einen Polynomialzeitverifizierer an. Beschreiben Sie dazu im Detail die Kodierung und die Länge des Zertifikates und die Arbeitsweise des Verifizierers.

> INDEPENDENTSET
>
> **Eingabe:** Ein Graph $G = (V, E)$ und eine Zahl $b \in \mathbb{N}$.
>
> **Ausgabe:** Ja, gdw. es eine Knotenmenge $K \subseteq V$ mit $|K| \geq b$ gibt, so dass es in E keine Kanten zwischen den Knoten aus K gibt.

Lösungsvorschlag.
Zertifikat und Länge: Das Zertifikat ist eine Teilmenge V' der Knoten von G, die ein Independent Set der Größe mindestens k bilden. Es sei $V = \{v_1, \ldots, v_n\}$. Die Kodierung des Zertifikats ist ein Bitstring $B = (b_1, \ldots, b_n) \in \{0,1\}^n$ der Länge $|V| = n$, wobei $b_i = 1 \Leftrightarrow v_i \in V'$ gilt.
Polynomialzeitverifizierer: Der Zertifizierer muss testen, ob das Zertifikat die richtige Form hat und ob es ein gültiges Independent Set kodiert.

(1) Gilt $B \in \{0,1\}^n$?

(2) Für alle $i \in \{1, \ldots, n\}$ mit $b_i = 1$ wird getestet, ob für alle Kanten der Form $(v_i, v_j) \in E$ der Knoten v_j nicht in V' ist, also ob $v_j \notin V'$ gilt.

Laufzeit: In Schritt 1 wird die gesamte Eingabe überprüft, was in Zeit $\mathcal{O}(n)$ erledigt wird. In Schritt 2 werden von jedem Knoten die Nachbarn untersucht. Ein Knoten kann maximal $n - 1$ viele Nachbarn haben, jeder dieser Nachbarn kann durch Betrachten der Adjazenzmatrix identifiziert werden. Diese Identifikation läuft in Zeit $\mathcal{O}(n^2)$ ab, bei maximal $n - 1$ vielen Nachbarn ergibt sich damit ein Aufwand von $\mathcal{O}(n^3)$. Da auf diese Art alle Knoten bearbeitet werden, kann Schritt 2 in Zeit $\mathcal{O}(n^4)$ abgearbeitet werden. Der Verifizierer läuft folglich in Polynomialzeit.

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte) Geben Sie zum folgenden Entscheidungsproblem ein Zertifikat und einen Polynomialzeitverifizierer an. Beschreiben Sie dazu im Detail die Kodierung und die Länge des Zertifikates und die Arbeitsweise des Verifizierers.

3SAT
Eingabe: Eine Aussagenlogische Formel φ in 3-KNF.
Ausgabe: Ja, gdw.es eine erfüllende Belegung der Variablen gibt.

Lösungsvorschlag.
Wähle eine erfüllende Variablenbelegung als Zertifikat.
Zertifikat: Ein Wort $y \in \{0,1\}^k$, wenn die Formel k Variablen nutzt (natürlich ist $k \leq n$). An Stelle i steht die Belegung von X_i.
Polynomialzeitverifizierer:

1. Prüfe, ob $y \in \{0,1\}^k$. (Vergleiche Anzahl Variablen mit Länge von y.)
2. Überprüfe, ob y eine erfüllende Variablenbelegung kodiert. Laufe dazu über die Formel und überprüfe, ob jede Klausel erfüllt ist.

Laufzeit: Für Schritt 1 muss einmal über die Formel und gelaufen werden und gleichzeit geschaut werden, dass y die richtige Länge hat. Das geht auf jeden Fall in $O(n^2)$ Schritten auf einer 1-Band-TM. Schritt 2 muss auch einmal über die Formel laufen und dabei jeweils die Werte von y nachschauen. Das geht auch in Zeit $O(n^2)$.

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)
Konstruieren Sie für folgende Sprache L eine NTM mit einem Band die L erkennt. Geben Sie auch die Laufzeit ihrer Konstruktion in O-Notation an.

$$L = \{v\#w \mid v, w \in \{0,1\}^* \text{ und } v \neq w\}$$

Lösungsvorschlag.
Wir konstruieren jetzt eine NTM mit einem Band und schätzen dann deren Laufzeit ab.

1. Prüfe, ob die Eingabe von der Form $v\#w$ für $v, w \in \{0,1\}^*$.
2. Laufe von links nach rechts über das Teilwort $v\#$ und zähle bis zu einer Position i in einer zweiten Spur die Anzahl der gelesenen Bits. Speichere den Buchstaben v_i im aktuellen Zustand. Die Position i wird nichtdeterministisch bestimmt, indem die Übergangsrelation an jeder Bandposition (innerhalb von $v\#$) die Wahl erlaubt, entweder weiterzulaufen oder in die nun folgende Phase überzugehen.
3. Laufe mit Hilfe des Zählers bis zur Position i hinter dem Zeichen $\#$, lese den Buchstaben w_i und vergleiche v_i und w_i.
   (1) Falls $v_i \neq \#$, $w_i \neq B$ und $v_i \neq w_i$ ist, dann unterscheiden sich v und w in diesem Buchstaben, $\Rightarrow v \neq w$.
   (2) Falls $v_i = \#$ ist und $w_i \neq B$, dann ist v ein echtes Präfix von w, $\Rightarrow v \neq w$.
   (3) Falls $v_i \neq \#$ ist und $w_i = B$, dann ist w ein echtes Präfix von v, $\Rightarrow v \neq w$.

Die Laufzeit beträgt $O(n \log n)$. Der Zähler benötigt Zeit $O(\log n)$, und wir müssen einmal über das Band laufen.

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)
Konstruieren Sie für folgende Sprache L eine NTM die L erkennt. Geben Sie auch die Laufzeit ihrer Konstruktion in O-Notation an.

$$L = \{v\#w \mid v, w \in \{0,1\}^* \text{ und } w \neq v^R\}$$

Lösungsvorschlag.
Wir konstruieren jetzt eine NTM mit einem Band und schätzen dann deren Laufzeit ab.

1. Prüfe, ob die Eingabe von der Form $v\#w$ für $v, w \in \{0,1\}^*$ und ob beide Wörter gleich lang sind, speichere dabei auch die Länge n des Wortes w auf der zweiten Spur.

2. Laufe von links nach rechts über das Teilwort $v\#$ und zähle bis zu einer Position i in einer weiteren Spur die Anzahl der gelesenen Bits. Nun kann auf der zweiten Spur die Differenz $n - i$ berechnet werden, dies ist nun unser Zähler. Speichere den Buchstaben v_i im aktuellen Zustand. Die Position i wird nichtdeterministisch bestimmt, indem die Übergangsrelation an jeder Bandposition (innerhalb von $v\#$) die Wahl erlaubt, entweder weiterzulaufen oder in die nun folgende Phase überzugehen.

3. Laufe mit Hilfe des Zählers bis zur Position $n - i$ hinter dem Zeichen $\#$, lese den Buchstaben w_{n-i} und vergleiche v_i und w_{n-i}.

   (1) Falls $v_i \neq \#$, $w_{n-i} \neq B$ und $v_i \neq w_{n-i}$ ist $\Rightarrow v \neq w^R$.

   (2) Falls $v_i = \#$ ist und $w_{n-i} \neq B$, dann ist v ein echtes Präfix von w, $\Rightarrow w \neq v^R$.

   (3) Falls $v_i \neq \#$ ist und $w_{n-i} = B$, dann ist w ein echtes Präfix von v, $\Rightarrow w \neq v^R$.

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

WS 2012/13
Blatt 10

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)
Geben Sie die Definition der Problem-Klasse NP an.

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)
Wann ist ein Problem A auf B reduzierbar? Geben Sie die Definition von polynomieller Reduktion an.

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)

Zeigen Sie, dass das folgende Entscheidungsproblem in NP ist.

DOMINATING SET

Eingabe: Ein Graph $G = (V, E)$ und eine Zahl $b \in \mathbb{N}$.

Ausgabe: Ja, gdw. es eine Knotenmenge $D \subseteq V$ mit $|D| \leq b$ gibt, so dass jeder Knoten in $V \setminus D$ zu einem Knoten in D adjazent ist.

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)
Zeigen Sie, dass das Boolean Circuit in NP ist.

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)

Geben Sie eine polynomielle Reduktion von VertexCover auf IndependentSet an, formell: VertexCover $\leq_p$ IndependentSet.

Zur Erinnerung hier die Definitionen der Probleme:

VertexCover

Eingabe: Ein Graph $G = (V, E)$ und eine Zahl $b \in \mathbb{N}$.

Ausgabe: Ja, gdw. es eine Knotenmenge $K \subseteq V$ mit $|K| \leq b$ gibt, so dass es jede Kante in E zu mindestens einem Knoten aus K inzident ist.

IndependentSet

Eingabe: Ein Graph $G = (V, E)$ und eine Zahl $b \in \mathbb{N}$.

Ausgabe: Ja, gdw. es eine Knotenmenge $K \subseteq V$ mit $|K| \geq b$ gibt, so dass es in E keine Kanten zwischen den Knoten aus K gibt.

RWTH Aachen **WS 2012/13**
Lehrgebiet Theoretische Informatik Blatt 11
Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)

Geben Sie eine polynomielle Reduktion von Clique auf IndependentSet an, formell: Clique $\leq_p$ IndependentSet.
Zur Erinnerung hier die Definitionen der Probleme:

Clique

Eingabe: Ein Graph $G = (V, E)$ und eine Zahl $b \in \mathbb{N}$.

Ausgabe: Ja, gdw. es eine Knotenmenge $K \subseteq V$ mit $|K| \leq b$ gibt, so dass für alle Knoten $a, b \in K$ paarweise durch eine Kante verbunden sind $(a, b) \in E$.

IndependentSet

Eingabe: Ein Graph $G = (V, E)$ und eine Zahl $b \in \mathbb{N}$.

Ausgabe: Ja, gdw. es eine Knotenmenge $K \subseteq V$ mit $|K| \geq b$ gibt, so dass es in E keine Kanten zwischen den Knoten aus K gibt.

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)

Geben Sie eine polynomielle Reduktion von PARTITION auf KNAPSACK an, formell: PARTITION $\leq_p$ KNAPSACK.
Zur Erinnerung hier die Definitionen der Probleme:

PARTITION

Eingabe: Eine Zahlenmenge $X \in \mathbb{N}$.

Ausgabe: Ja, gdw. $X = L \cup R$ so in zwei disjunkte Mengen zerlegt werden kann $L \cap R = \emptyset$, dass gilt $\sum_{x_i \in L} = \sum_{x_j \in R}$.

KNAPSACK

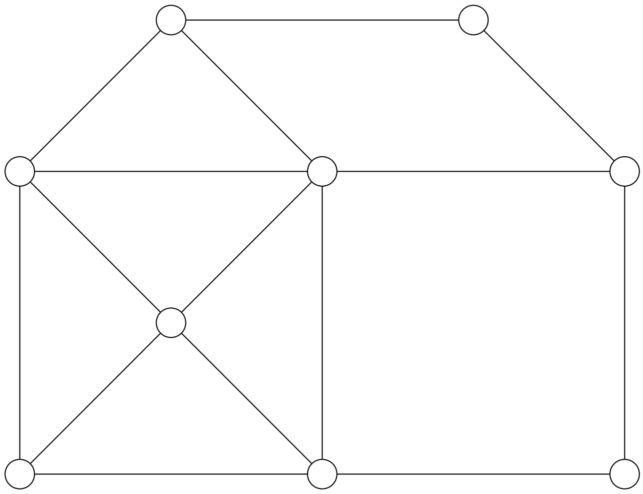
Eingabe: Eine Menge von Items I, wobei jedes Item $i \in I$ ein Größe $w_i \in \mathbb{N}$ und einen Wert $v_i \in \mathbb{N}$ hat. Außerdem eine Gewichtsschranke $w \in \mathbb{N}$ und eine Wertschranke $v \in \mathbb{N}$.

Ausgabe: Ja, gdw. es eine Teilmenge der Items $I' \subseteq I$ gibt, sodass $\sum_{i' \in I'} w_i \leq w$ und $\sum_{i' \in I'} v_i \geq v$.

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)

Sei der abgebildete Graph gegeben. Geben sie eine minimale Färbung des Graphen (z.B. durch ausmalen oder beschriften der Knoten) an.



Bonusfrage: Hat der Graph einen Hamilton-Pfad?

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

WS 2012/13
Blatt 11

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)

Ist die folgende Instanz von SAT erfüllbar? Falls ja, geben sie eine Lösung an. Ansonsten begründen Sie, warum die Instanz unerfüllbar ist.

$$(\overline{a} \vee \overline{b}) \wedge (a \vee \overline{b} \vee c) \wedge (a \vee \overline{b} \vee \overline{c}) \wedge (\overline{a} \vee b \vee c) \wedge (a \vee b)$$

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe 1 (10 Punkte)

Ist das beschriebene Problem polynomiell lösbar oder nicht? Geben Sie einen Polynomialzeitalgorithmus oder einen NP-härte Beweis an.

3-Partition

Eingabe: Gegeben ist eine Menge von Zahlen $A = \{ a_1, a_2, \ldots, a_n \}$.

Ausgabe: Ja, gdw. es drei disjunkte Teilmengen $A' \cup A^* \cup A^+ = A$ gibt, so dass $\sum_{w' \in A'} w' = \sum_{w^* \in A^*} w^* = \sum_{w^+ \in A^+} w^+$.

Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe (10 Punkte)

Ist das beschriebene Problem polynomiell lösbar oder nicht? Geben Sie einen Polynomialzeitalgorithmus oder einen NP-härte Beweis an.

WeightedVertexCover

Eingabe: Ein Graph $G = (V, E, w)$ mit $w\colon V \to \mathbb{N}, v \mapsto w(v)$ und eine Zahl $b \in \mathbb{N}$.

Ausgabe: Ja, gdw. es eine Knotenmenge $K \subseteq V$ mit $\sum_{v\in K} w(v) \leq b$ gibt, so dass jede Kante in E durch mind. einen Knoten abgedeckt ist.

RWTH Aachen **WS 2012/13**
Lehrgebiet Theoretische Informatik Blatt 12
Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe (10 Punkte)

Eine bekannte Kaffeehaus-Kette möchte in Aachen im großen Stil Filialen eröffnen. Der Plan sieht vor Filialen so an Straßenkreuzungen zu platzieren, dass von jeder Straße aus mindestens ein Kaffeehaus sichtbar ist. Offensichtlich wird hierzu eine große Anzahl an Neueröffnungen benötigt und das Unternehmen möchte, dass Sie die Anzahl der benötigten Kaffeehäusern minimieren.

Ist das beschriebene Problem polynomiell lösbar oder nicht? Geben Sie einen Polynomialzeitalgorithmus oder einen NP-härte Beweis an.

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe (10 Punkte)

Ist das beschriebene Problem polynomiell lösbar oder nicht? Geben Sie einen Polynomialzeitalgorithmus oder einen NP-härte Beweis an.

Sie haben die Zulassung in Berechenbarkeit und Komplexität erhalten und wollen dies mit einer Party feiern. Dazu wollen Sie so viele Freunde wie möglich einladen. Leider mögen sich Ihre Freunde untereinander teilweise nicht besonders, so dass es für eine stimmungsvolle Part besser ist, keine zwei Personen einzuladen die sich nicht mögen.

Ist das Problem eine möglichst große Gästeliste zusammenzustellen polynomiell lösbar oder nicht? Geben Sie einen Polynomialzeitalgorithmus oder einen NP-härte Beweis an.

RWTH Aachen
Lehrgebiet Theoretische Informatik
Reidl–Ries–Rossmanith–Sanchez–Tönnis

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe (10 Punkte)

Sie haben die Zulassung in Berechenbarkeit und Komplexität erhalten und wollen dies mit einer großen Party feiern. Die Einladung soll an alle ihre Freunde und deren Freunde gehen. Leider haben Sie keine Kontaktdaten vom Großteil der potentiellen Gäste. Zum Glück wissen Sie, wer mit wem befreundet ist und können damit ein Bekanntschaftsnetzwerk erstellen. Dabei werden Gäste durch Knoten und Freundschaften durch Kanten repräsentiert. Sie schätzen, dass es ungefähr einen Tag dauert, bis ein Gast die Einladung an seine Freunde weitergeleitet hat. Wie viele Tage im voraus müssen sie ihre Einladung versenden, damit alle Gäste rechtzeitig informiert werden?

Ist das beschriebene Problem polynomiell lösbar oder nicht? Geben Sie einen Polynomialzeitalgorithmus, und dessen Laufzeit, oder einen NP-härte Beweis an.