

Einführung in die Theoretische Informatik

Klausur, Haupttermin SoSe 2021, 19. Juli 2021

Aufgabe 1: Informationstheorie

(10 Punkte)

(a) **Erwartete Codewortlänge**

(4 Punkte)

Betrachten Sie die nebenstehende Tabelle für die Quelle (Σ, p) .

Ist der gegebene Code \mathbb{C} über dem Alphabet $\{0, 1, 2\}$ ein Präfixcode minimaler erwarteter Codewortlänge? Begründen Sie.

σ	a	b	c	d	e	f	g
p_σ	$\frac{1}{18}$	$\frac{4}{18}$	$\frac{2}{18}$	$\frac{1}{18}$	$\frac{3}{18}$	$\frac{3}{18}$	$\frac{4}{18}$
$\mathbb{C}(\sigma)$	22	21	10	11	00	01	20

(b) **Erwarteter Informationsgewinn**

(6 Punkte)

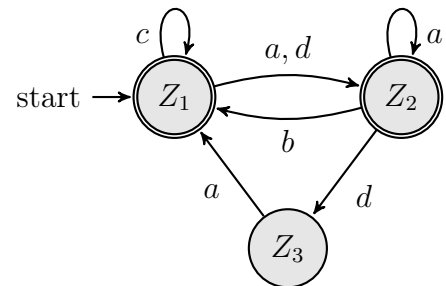
Sei $\varphi: \Sigma^* \rightarrow \{a, b, c, d\}^*$ eine *Filterfunktion*, welche die Zeichen e, f und g durch ε ersetzt und die übrigen Zeichen unverändert lässt. Sei m eine Nachricht, deren Zeichen jeweils unabhängig aus der oben angegebenen Quelle (Σ, p) stammen.

Bestimmen Sie den erwarteten Informationsgewinn der Nachricht $\varphi(m)$ in Abhängigkeit von $k := |\varphi(m)|$. Vereinfachen Sie Ihr Ergebnis, bis es keine log-Terme mehr enthält.

Aufgabe 2: Umwandlung DEA \rightarrow RegEx

(8 Punkte)

Gegeben sei der rechts abgebildete DEA. Wandeln Sie ihn – gemäß dem Vorgehen aus der Vorlesung! – in einen regulären Ausdruck um.



Aufgabe 3: Pumping Lemma

(10 Punkte)

(a) **Wortmindestlänge**

(2 Punkte)

Betrachten Sie die reguläre Sprache $A := \mathcal{L}(a^+b^*)$. Geben Sie die *minimale* Wortmindestlänge $n(A)$ an, sodass eine Zerlegung gemäß dem Pumping Lemma existiert.

(b) **Anwendung**

(8 Punkte)

Zeigen Sie mithilfe des Pumping Lemmas, dass

$$L := \{b^j a^k b^\ell \mid j \bmod 3 = 2, (k + \ell) \bmod 3 = 0, j + \ell \leq k\}$$

nicht regulär ist.

Aufgabe 4: Rechnende Turingmaschine**(10 Punkte)**

Sie kennen die Binärkodierung $\mathbb{B}(\alpha)$ einer natürlichen Zahl α . Analog sei $\mathbb{Q}(\alpha)$ die *Quaternärkodierung* von α , d. h. ihre Kodierung zur Basis 4 mithilfe der Symbole $\{0, 1, 2, 3\}$.

Erstellen Sie eine deterministische TM mit Bandalphabet $\Gamma = \{0, 1, 2, 3, \$, \square\}$, die bei Eingabe $\mathbb{B}(\alpha)$ die folgende Funktion berechnet und **in Quaternärkodierung** ausgibt:

$$f(\alpha) := \begin{cases} \alpha + 1, & \text{wenn } \alpha \bmod 4 \neq 3; \\ \text{undef}, & \text{sonst.} \end{cases}$$

Beispiel: Für die Dezimalzahl 23 ist $\mathbb{Q}(23) = 113$, denn $23 = 1 \cdot 4^2 + 1 \cdot 4^1 + 3 \cdot 4^0$.

Hinweis: Beachten Sie, dass die Fallunterscheidung die Addition für Sie vereinfacht.

Aufgabe 5: Turing-Vollständigkeit**(12 Punkte)**

Eine *2-nachlaufende Turingmaschine* ist eine rechnende Turingmaschine mit folgenden Modifikationen:

- Anstelle des Schreib-Lese-Kopfes gibt es einen Schreibkopf und einen Lesekopf.
- Der Schreibkopf steht 2 Zellen weiter links als der Lesekopf.
- Die Angabe der Bewegungsrichtung bei einem Übergang gilt für beide Köpfe gleichzeitig.
- Der Schreibkopf muss nicht bei jedem Übergang etwas schreiben; in diesem Fall wird anstelle des zu schreibenden Symbols am Übergang ε notiert.
- Zu Beginn bzw. Ende der Ausführung steht der Lesekopf auf dem ersten Symbol der Eingabe bzw. Ausgabe.

(a) Beweis**(8 Punkte)**

Zeigen Sie, dass 2-nachlaufende Turingmaschinen mindestens genauso mächtig sind wie rechnende Turingmaschinen.

(b) Halteproblem**(4 Punkte)**

Ist das folgende Problem entscheidbar? Begründen Sie.

2-NACHHALT

Gegeben: Eine 2-nachlaufende TM \mathcal{M} mit Bandalphabet $\Gamma := \{\square, 0, 1, 2\}$.

Gefragt: Hält \mathcal{M} bei Eingabe 2?

Aufgabe 6: Berechenbarkeit und Entscheidbarkeit**(14 Punkte)****(a) Berechenbarkeit****(6 Punkte)**

Begründen Sie für die folgende Funktion (mit Eingabe $\alpha \in \mathbb{N}$), ob sie berechenbar ist:

$$f(\alpha) := \begin{cases} \text{undef,} & \text{falls } \alpha \text{ die Kodierung } \mathbb{W}(\mathcal{M}) \text{ einer TM } \mathcal{M} \text{ ist, die bei Eingabe } \alpha \\ & \text{nicht terminiert oder eine Ausgabe der Länge 7 hat;} \\ 1, & \text{sonst.} \end{cases}$$

(b) Semi-Entscheidbarkeit**(8 Punkte)****HEIMWEH****Gegeben:** Rechnende Turingmaschine \mathcal{M} .**Gefragt:** Gibt es ein $w \in \Sigma^*$, sodass der Schreib-/Lesekopf von \mathcal{M} bei Eingabe w die Zelle des Speicherbandes, auf der er gestartet ist, mindestens dreimal besucht?

Ist HEIMWEH semi-entscheidbar? Begründen Sie (ggf. genügt Pseudocode).

Aufgabe 7: Komplexität**(14 Punkte)****(a) Pseudopolynomieller Algorithmus****(8 Punkte)****ZAHLMENGE****Gegeben:** Endliche Menge natürlicher Zahlen $M \subseteq \mathbb{N}$.**Gefragt:** Gibt es eine Zahl in M , die keine Primzahl ist?

Geben Sie einen deterministischen Algorithmus an, der ZAHLMENGE in **pseudopolynomieller Zeit** entscheidet, ohne einen Primzahltest als Blackbox zu verwenden. Begründen Sie.

Hinweis: Nehmen Sie an, dass Sie für zwei natürliche Zahlen $\alpha < \beta$ in Zeit $\mathcal{O}(\log \beta)$ entscheiden können, ob β von α geteilt wird.

(b) Fixed-Parameter Tractability**(6 Punkte)**

Sei \mathcal{Y} ein Entscheidungsproblem, dessen Eingabe aus einem Graphen (ohne Mehrfachkanten) besteht. Zeigen Sie:

$\mathcal{Y} \in \mathbf{FPT}$ in Bezug auf die Knotenanzahl des Eingabegraphen $\iff \mathcal{Y}$ ist entscheidbar.

Aufgabe 8: NP-Vollständigkeit**(16 Punkte)**

Ein Rechteck $R = (a, b)$ ist gegeben durch seine beiden Kantenlängen $a, b \in \mathbb{N}$. Ein Rechteck heißt *nicht-trivial*, wenn beide Kanten mindestens Länge 2 haben. Der Flächeninhalt eines Rechtecks $R = (a, b)$ berechnet sich als $a \cdot b$.

FÜNFTELFLÄCHE

Gegeben: Ein nicht-triviales Rechteck $Z = (a, b)$; Menge $\mathcal{R} = \{R_1, \dots, R_n\}$ von nicht-trivialen Rechtecken mit $R_i = (a_i, b_i)$.

Gefragt: Gibt es eine Auswahl $S \subseteq \mathcal{R}$ von nicht-trivialen Rechtecken, sodass diese überlappungsfrei zu einem Rechteck angeordnet werden können, dessen Flächeninhalt genau ein Fünftel des Flächeninhalts von Z ist?

(a) Reduktion**(12 Punkte)**

Zeigen Sie, dass FÜNFTELFLÄCHE **NP**-schwer ist.

Das in Ihrer Reduktion verwendete Ausgangsproblem \mathcal{X} muss aus der Vorlesung stammen. Geben Sie die Definition von \mathcal{X} nach dem bekannten Schema (Gegeben/Gefragt) an.

(b) Komplexität**(4 Punkte)**

Tatsächlich ist FÜNFTELFLÄCHE auch **NP**-vollständig. Angenommen es existiert ein deterministischer Algorithmus \mathcal{A} , der FÜNFTELFLÄCHE in Polynomialzeit korrekt entscheidet. Gilt dann $P = NP$? Begründen Sie.

Aufgabe 9: Randomisierte Algorithmen**(10 Punkte)**

In einer mündlichen Onlineprüfung (ohne Videoübertragung) sind für jede Prüfungsfrage exakt vier Minuten für das Stellen der Frage und das Geben der Antwort vorgesehen. Für die zu prüfende Person gilt, dass sie (falls die Audioverbindung funktioniert) auf jede Frage mit einer unabhängigen Wahrscheinlichkeit $p > 2/3$ eine Antwort gibt (egal ob richtig oder falsch).

Es kann jedoch sein, dass die Audioverbindung nicht korrekt funktioniert – dann wird für keine der Prüfungsfragen eine Antwort übertragen und die Prüfung ist *zu annullieren*. Der Zustand der Audioverbindung steht vor Beginn der Prüfung fest und ändert sich dann nicht mehr.

(a) Algorithmus**(4 Punkte)**

Stellen Sie sich vor, dass Sie als Beisitzende*r eine einzelne Prüfung protokollieren müssen. Geben Sie einen **RP**-Algorithmus für die Frage „Ist die Prüfung nicht zu annullieren?“ an.

(b) Analyse**(6 Punkte)**

Sei F die Anzahl der Prüfungsfragen. Zeigen Sie, welche Fehlerwahrscheinlichkeiten Ihr Algorithmus (in Abhängigkeit von F) auf *JA*- und welche er auf *NEIN*-Instanzen hat.