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

WS 2012/13
Übungsblatt 8
3.12.2012

Übung zur Vorlesung Berechenbarkeit und Komplexität

Aufgabe T18

a) Wiederholen Sie die Definition der O-Notation.

b) Welche der folgenden Aussagen sind wahr?

1. $n^2 = O(2^n)$
2. $\sqrt{n} = O(n/\log n)$
3. $\sqrt{n-1} = \sqrt{n} + O(1)$
4. $\displaystyle\int_0^{O(n)} O(t)\,dt = O(n^2)$
5. $(\log n)^n = O\big(n^{\log n}\big)$

Aufgabe T19

Wir bezeichnen mit p_N die N-te Primzahl.
Zu gegebener Eingabe $N \in \mathbb{N}$ sei die kleinste Primzahl gesucht, welche größer als N ist.
Ist diese Aufgabe in polynomieller Zeit lösbar, falls

a) N in unärer Kodierung vorliegt,

b) N in binärer Kodierung vorliegt?

Hinweis zu Teilaufgabe b: Cramérs Vermutung von 1936 besagt, daß die Primzahllücke $g(p_n) = p_{n+1} - p_n$ zwischen der n-ten und $(n+1)$-ten Primzahl durch

$$g(p_n) = O\big((\ln p_n)^2\big)$$

beschränkt ist.

Aufgabe H18 (5 Punkte)

Für einen beliebigen Algorithmus sei $t(n) \geq n$ die Laufzeit im uniformen Kostenmaß und $t'(n)$ die Laufzeit im logarithmischen Kostenmaß.
Finden Sie eine möglichst interessante Beziehung zwischen $t(n)$ und $t'(n)$.

Aufgabe H19 (10 Punkte)

Welche Funktion berechnet der folgende Algorithmus?

Analysieren Sie seine Laufzeit sowohl im uniformen als auch im logarithmischen Kostenmaß.

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

$b = 2$
while $a \neq 0$
  $b = b \cdot b$
  $a = a - 1$
end while
$c = 2$
while $b \neq 0$
  $c = c \cdot c$
  $b = b - 1$
end while

Ausgabe: c

Aufgabe H20 (8 Punkte)

Als Eingabe sei eine Menge von Intervallen

$$\left\{ \left[\frac{a_1}{b_1}, \frac{c_1}{d_1}\right], \left[\frac{a_2}{b_2}, \frac{c_2}{d_2}\right], \left[\frac{a_3}{b_3}, \frac{c_3}{d_3}\right], \dots, \left[\frac{a_n}{b_n}, \frac{c_n}{d_n}\right] \right\}$$

gegeben.

Die Anfangs- und Endpunkte jedes Intervalls sind rationale Zahlen, welche durch die Binärdarstellung ihres Zählers und Nenners kodiert werden.

Ist in polynomieller Zeit berechenbar, ob der Schnitt zwischen allen gegebenen Intervallen leer ist?

Falls Sie der Meinung sind, daß die Antwort *ja* ist, dann geben Sie einen Algorithmus an, der die Aufgabe in polynomieller Zeit löst. Wie groß ist die Laufzeit Ihres Algorithmus?

Aufgabe H21 (Bonusaufgabe)

Gegeben sind zwei Mengen von natürlichen Zahlen $\{k_1, \dots, k_n\}$ und $\{l_1, \dots, l_n\}$, die jeweils als durch Trennzeichen separierte Binärdarstellungen als Eingabestring repräsentiert werden.

Kann die folgende Frage in polynomieller Zeit beantwortet werden?

$$\sqrt{k_1} + \dots + \sqrt{k_n} < \sqrt{l_1} + \dots + \sqrt{l_n}$$

Hinweis: Vorsicht! Dies ist eine seit vielen Jahren offene Frage, deren Beantwortung weitreichende Konsequenzen hätte. Es ist heute noch nicht einmal bekannt, ob dieses Problem in *NP* liegt.