Tentamen[1]
Berekenbaarheid en Complexiteit (211170)

9 november 2007, 9:00 – 12:30 uur

N.B. (1) Tijdens het tentamen is het gebruik van het boek van Th.A. Sudkamp alsmede kopieën van overhead-sheets en artikelen, mits opgeborgen in een klapper (dus *geen losse blaadjes*), toegestaan.
(2) Een puntenverdeling, die de onderlinge zwaarte van vraagstukken en hun onderdelen aangeeft, bevindt zich in onderstaande tabel.

Opgave	(a)	(b)	(c)	(d)	(e)	Totaal
1	9	3	5	9	5	31
2	4	8	8	8	5	33
3	4	7	4			15
4	4	4	6	7		21
Totaal generaal						100

Opgave 1.

(a) Construeer een macro-grammatica $G_1 = (N, \Sigma, V, P, S)$ voor de taal L_1 over het alfabet $\Sigma = \{a, b, c\}$ gedefinieerd door

$$L_1 = \{xc^n x^R \mid x \in \{a,b\}^+,\ n = |x|\ \}$$

waarin $|x|$ de lengte van het woord x en R de spiegelingsoperatie (Eng. mirror, reversal) is. Beargumenteer dat de taal gegenereerd door G_1 inderdaad gelijk is aan L_1 en dat er dus geen sprake is van overgeneratie (dat wil zeggen $L_1 \subset L(G_1)$) of ondergeneratie (dat wil zeggen $L(G_1) \subset L_1$). **N.B.** "$\subset$" betekent echte inclusie.

(b) Is uw grammatica G_1 een OI-macro-grammatica of een IO-macro-grammatica of beide? Motiveer uw antwoord.

(c) Geef een afleiding volgens G_1 van de woorden $a^2bc^3ba^2$ en $b^2ac^3ab^2$ uit L_1.

(d) Beschouw vervolgens de taal L_0 over $\{a, b, c\}$ gedefinieerd door

$$L_0 = \{w^k \mid k = 2^m,\ m \geq 0,\ w \in L_1\}.$$

N.B. Er geldt $L_0 \subset L_1^\star$ en uiteraard $L_0 \neq L_1^\star$.
Construeer een macro-grammatica $G_0 = (N_0, \Sigma, V_0, P_0, S_0)$ voor de taal L_0.

(e) Is uw grammatica G_0 een OI-macro-grammatica of een IO-macro-grammatica of beide? Motiveer uw antwoord.

[1]On pp. 4–6 there is an English version.

Opgave 2.

Gegeven twee talen L_1 ($L_1 \subseteq \Sigma_1^\star$) en L_2 ($L_2 \subseteq \Sigma_2^\star$). De taal L_1 heet *reduceerbaar in polynomiale tijd* tot de taal L_2 —notatie: $L_1 \leq_p L_2$— als er een totale functie $f : \Sigma_1^\star \to \Sigma_2^\star$ bestaat met

(1) voor alle $x \in \Sigma_1^\star$: $\quad x \in L_1 \iff f(x) \in L_2$, en

(2) f is in polynomiale tijd berekenbaar, dat wil zeggen: er bestaat een deterministische k-band Turing-machine M ($k \geq 1$) die, gestart met x op band 1, $f(x)$ berekent, het antwoord $f(x)$ op band 1 afdrukt, en voldoet aan de eis: $tc_M(n) \in O(n^d)$ voor een of andere $d \geq 1$. Vergelijk Definitie 15.3.1 op p. 459 in [Sudkamp] 2nd ed[2].

(a) Toon aan dat de relatie $\leq_p$ reflexief en transitief is.

Zij $\mathcal{NP}$-SPACE de familie van talen, die door niet-deterministische Turing-machines worden geaccepteerd in polynomiale ruimte. Een taal L_0 noemen we $\mathcal{NP}$-SPACE-*volledig* als:

(1) $L_0 \in \mathcal{NP}$-SPACE, en

(2) $\forall L \in \mathcal{NP}\text{-SPACE} : \; L \leq_p L_0$.

(b) Bewijs dat als L_1 $\mathcal{NP}$-SPACE-volledig is, $L_2 \in \mathcal{NP}$-SPACE en $L_1 \leq_p L_2$, dan is ook L_2 $\mathcal{NP}$-SPACE-volledig.

(c) Toon dat als voor een $\mathcal{NP}$-SPACE-volledige taal L_1 geldt dat $L_1 \in \mathcal{NP}$, dat dan $\mathcal{NP} = \mathcal{NP}$-SPACE.

Zij *co*-$\mathcal{NP}$-SPACE gedefinieerd door

$$\textit{co-}\mathcal{NP}\text{-SPACE} = \{\Sigma^\star - L \mid L \subseteq \Sigma^\star,\ L \in \mathcal{NP}\text{-SPACE}\}.$$

(d) Bewijs dat als L een $\mathcal{NP}$-SPACE-volledige taal is met $L \in$ *co*-$\mathcal{NP}$-SPACE, dan geldt $\mathcal{NP}$-SPACE $=$ *co*-$\mathcal{NP}$-SPACE.

(e) Kunt u de gelijkheid $\mathcal{NP}$-SPACE $=$ *co*-$\mathcal{NP}$-SPACE ook bewijzen zonder de aanname "als L een $\mathcal{NP}$-SPACE-volledige taal is met $L \in$ *co*-$\mathcal{NP}$-SPACE"? En hoe?

Opgave 3.

Beschouw het volgende probleem genaamd **P3**.

P3: Gegeven een eindige lijst van 3-tallen van binair gerepresenteerde natuurlijke getallen $(a_1, b_1, c_1), (a_2, b_2, c_2), \ldots, (a_n, b_n, c_n)$ met $n \geq 2$ waarbij $a_i, b_i, c_i \geq 1$ ($1 \leq i \leq n$). Bestaat er een deelverzameling D van $\{1, 2, \ldots, n\}$ zodanig dat

$$\sum_{i \in D} a_i \; = \; \sum_{i \notin D} (b_i + c_i) \;\; ?$$

(a) Toon schetsmatig aan dat **P3** tot $\mathcal{NP}$ behoort.

(b) Toon aan dat **P3** ook $\mathcal{NP}$-volledig is.

(c) Is **P3** ook sterk $\mathcal{NP}$-volledig? Beargumenteer uw antwoord.

[2] Dit is Definitie 15.6.1 op p. 477 in [Sudkamp] 3rd ed.

Opgave 4.

Gegeven zijn twee recursief opsombare (Eng. recursively enumerable) talen L_1 en L_2 over het alfabet Σ; dus er bestaan Turing-machines M_1 en M_2 die respectievelijk L_1 en L_2 accepteren. [Bedenk daarbij steeds dat *niet* gegarandeerd is, dat deze Turing-machines voor elke invoer stoppen.] Daarnaast is er een eindige taal F gegeven met $F \subseteq \Sigma^\star$.

(a) Geef in globale termen een beschrijving van een Turing-machine M die de taal $L_1 \backslash L_2 = \{y \mid \exists x \in L_1 : xy \in L_2 \}$ accepteert; met andere woorden: toon aan dat ook $L_1 \backslash L_2$ recursief opsombaar is. [Aanwijzing. Met "in globale termen" wordt het volgende bedoeld: het is niet de bedoeling dat u in detail de (niet)deterministische Turing-machine definieert maar u mag globale instructies als "start de berekening van de invoer x op M_1 en als M_1 stopt dan schrijven we xx op de tweede band van M en starten M_2 op de inhoud van deze tweede band" of "we kopiëren de invoer x naar band 2 en naar band 3, doen daarna eerst 3 rekenstappen van M_1 op de inhoud van band 2, dan 4 rekenstappen van M_2 op de inhoud van band 3, vervolgens weer 2 stappen van M_1 op band 2 en daarna $\cdots$" gebruiken.]

(b) Geef in globale termen een beschrijving van een Turing-machine M_F die de taal $L_1 - F = \{x \in L_1 \mid x \notin F\}$ accepteert; met andere woorden: toon aan dat ook $L_1 - F$ recursief opsombaar is.

(c) Als nu bovendien L_1 en L_2 recursief (of beslisbaar) zijn, bewijs dan dat ook $L_1 \cap L_2$ recursief (beslisbaar) is.

(d) We keren terug naar het geval dat L_1 en L_2 recursief opsombaar zijn. Bewijs dat als $L_1 \cup L_2$ recursief is en $L_1 \cap L_2 = F$ geldt, dat dan beide talen L_1 en L_2 recursief zijn.