DEELTENTAMEN INLEIDING FUNDAMENTELE INFORMATICA
Donderdag 28 februari 2002, 13.45 u. - 15.30 u.

In totaal zijn er 100 punten te verdienen. Lees voor je begint alle opgaven goed door; maak de onderdelen die je gelijk kan beantwoorden het eerst.

Opgave 1 [25 pt]
Gegeven is het alfabet $\Sigma = \{a, b, c, d\}$.
Voor elke $i \in \{0, 1, 2, 3, 4\}$ is L_i de taal bestaande uit alle woorden waarin precies i verschillende symbolen voorkomen; dus $L_i = \{x \in \Sigma^* : \#alph(x) = i\}$.

(a) Geef één woord uit L_0, twee uit L_1 en drie uit L_2.

(b) Zijn de volgende uitspraken waar of niet waar? Waarom?
$L_0 \cdot L_1 = L_1$, $\quad L_1 \cdot L_2 = L_2$, $\quad L_2 \cdot L_3 \subseteq L_4$, $\quad L_4 \subseteq L_2 \cdot L_3$, $\quad (L_2 \cdot L_3) \cap L_4 \neq \emptyset$.

(c) Laat zien dat $\Sigma^* = L_0 \cup L_1 \cup L_2 \cup L_3 \cup L_4$.

(d) Definieer het begrip aftelbaarheid.

(e) Laat zien dat L_1 aftelbaar is.

(f) Zijn L_1 en Σ^* gelijkmachtig? Leg uit.

Opgave 2 [20 pt]
De functie $f : \{a, b\}^* \to \{a, b\}^*$ levert voor elk woord in $\{a, b\}^*$ zijn gesorteerde versie op. f wordt inductief gedefinieerd door $f(\lambda) = \lambda$
en $f(ax) = a \cdot f(x)$ en $f(bx) = f(x) \cdot b$ voor alle $x \in \{a, b\}^*$.

(a) Bereken: $f(b)$, $\quad f(ab)$, $\quad f(ba)$, $\quad f(bbab)$

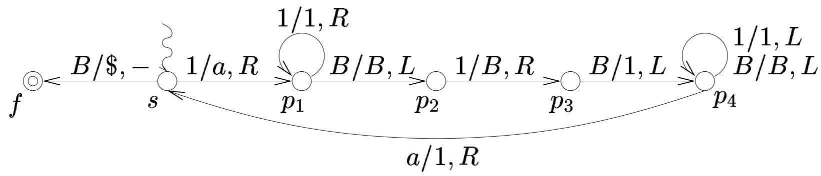
(b) Geef alle woorden $u \in \{a, b\}^*$ zodat $f(u) = a^2 b$.
Geef alle woorden $u \in \{a, b\}^*$ zodat $f(u) = b^2 a$.

(c) Is f totaal, injectief, surjectief? Waarom (niet)?

(d) Bewijs met inductie naar $|w|$: voor alle $w \in \{a, b\}^*$ is $f(w) = a^{|w|_a} b^{|w|_b}$.

Ter herinnering: voor een woord y en een letter d is $|y|_d$ het aantal malen dat d voorkomt in y.

Opgave 3 [15 pt]
Gegeven is het toestandsdiagram van de Turingmachine M met invoeralfabet $\{1\}$ en bandalfabet $\{1, B, a, \$\}$:



(a) Geef de definities van
$L(M)$, de taal geaccepteerd (herkend) door M en
f_M, de functie berekend door M.

(b) Laat met behulp van configuratierijen zien dat λ en 11 worden geaccepteerd door M.
Geef $f_M(\lambda)$ en $f_M(11)$.

(c) Geef 2 woorden uit $\{1\}^*$ die niet worden geaccepteerd door M en leg uit waarom ze niet worden geaccepteerd.

Opgave 4 [`20 pt`]

(a) Definieer
$\mathcal{L}_{DTM}$, de familie van herkenbare (DTM) talen, en
$\mathcal{L}_{REC}$, de familie van beslisbare (recursieve) talen.

Gegeven is dat de Turingmachine M uit opgave 3 voor geen enkel invoerwoord een oneindige berekening heeft. Dus voor elk woord $x \in \{1\}^*$ zal de berekening van M op x na eindig veel rekenstappen (succesvol) stoppen of hangen.

(b) Construeer uit M een deterministische Turingmachine $\overline{M}$ die het complement van M accepteert. Dus $L(\overline{M}) = \{1\}^* - L(M)$.

(c) Is $L(M)$ een taal uit $\mathcal{L}_{REC}$? Leg uit.

Opgave 5 [`20 pt`]
Maak een DTM M met invoeralfabet $\{1\}$ die herkent of een (unair gerepresenteerd) getal een macht van 2 is. Dus

$$L(M) = \{1^k : k = 2^n \text{ voor een } n \in \mathbb{N}_0\}.$$

Leg het idee achter je constructie uit en geef een duidelijke toelichting op het gebruikte algoritme.

`einde deeltentamen`

UITWERKINGEN
DEELTENTAMEN INLEIDING FUNDAMENTELE INFORMATICA
Donderdag 28 februari 2002, 13.45 u. - 15.30 u.

Opgave 1

(a) $\lambda \in L_0$; $a \in L_1$, $b \in L_1$; $ab \in L_2$, $ba \in L_2$, $bc \in L_2$.
NB: $L_0 = \{\lambda\}$ en bevat dus maar één woord, maar zowel L_1 als L_2 zijn oneindig en voor deze talen zijn dus veel verschillende antwoorden goed!
$L_1 = \{a\}^+ \cup \{b\}^+ \cup \{c\}^+ \cup \{d\}^+$ en $L_2 = \{a,b\}^*\{ab,ba\}\{a,b\}^* \cup \{b,c\}^*\{bc,cb\}\{b,c\}^* \cup \{c,d\}^*\{cd,dc\}\{c,d\}^* \cup \{d,a\}^*\{da,ad\}\{d,a\}^*$.

(b) $L_0 \cdot L_1 = L_1$ is waar: $L_0 = \{\lambda\}$ en voor alle woorden $x \in L_1$ geldt $\lambda \cdot x = x$.
$L_1 \cdot L_2 = L_2$ is niet waar: $a \in L_1$ en $bc \in L_2$, maar niet $abc \in L_2$.
of: in alle woorden van L_1 komt precies één letter minstens één keer voor, dus alle woorden van L_1 hebben minstens lengte 1; op dezelfde manier weet je dat alle woorden uit $L-2$ minstens lengte 2 hebben. Dus alle woorden uit $L_1 \cdot L_2$ hebben minstens lengte 3. Het woord ab is dus wel een element van L_2 maar niet van $L_1 \cdot L_2$.
$L_2 \cdot L_3 \subseteq L_4$ is niet waar:
$ab \in L_2$ en $abc \in L_3$, dus $ababc \in L_2 \cdot L_3$. Omdat $alph(ababc) = \{a,b,c\}$ uit 3 letters bestaat en niet uit 4 is $ababc \notin L_4$.
$L_4 \subseteq L_2 \cdot L_3$ is niet waar:
$abcd \in L_4$, terwijl alle woorden uit $L_2 \cdot L_3$ volgens het ook al boven gegeven lengte argument minstens lengte 5 hebben.
of: als $abcd \in L_2 \cdot L_3$ dan moeten er $x \in L_2$ en $y \in L_3$ zijn zodat $abcd = xy$. Neem aan dat zo'n x en y bestaan. Omdat x uit precies twee verschillende letters bestaat, moet wel $x = ab$ gelden. Op dezelfde manier, Omdat y uit precies drie verschillende letters bestaat, moet wel $y = bcd$ gelden. Maar dan is $xy = abbcd \neq abcd$ (tegenspraak). Dus $abcd \notin L_2 \cdot L_3$.
$L_2 \cdot L_3 \cap L_4 \neq \emptyset$ is waar: $abbcd \in L_2 \cdot L_3 \cap L_4$.

(c) Gegeven is dat L_0, L_1, L_2, L_3 en L_4 alle vijf deelverzamelingen zijn van Σ^*. Daaruit volgt onmiddellijk dat $L_0 \cup L_1 \cup L_2 \cup L_3 \cup L_4 \subseteq \Sigma^*$.
Andersom, bekijk een willekeurig woord $x \in \Sigma^*$. Omdat Σ uit 4 verschillende symbolen bestaat, volgt dat in x hooguit 4 verschillende letters kunnen voorkomen. Dus $\#alph(x) = i$, met $0 \leq i \leq 4$, en dus is $x \in L_i$, wat impliceert dat $x \in L_0 \cup L_1 \cup L_2 \cup L_3 \cup L_4$. Conclusie: $\Sigma^* \subseteq L_0 \cup L_1 \cup L_2 \cup L_3 \cup L_4$.

(d) Een verzameling is aftelbaar als hij eindig is of gelijkmachtig met $\mathbb{N}$.

(e) Er zijn verschillende manieren om te laten zien dat L_1 aftelbaar is.
Je kan bijv. een beroep doen op de theorie uit het (werk)college:
L_1 is een deelverzameling van de aftelbare verzameling Σ^* en deelverzamelingen van aftelbare verzamelingen zijn zelf ook aftelbaar.
Of je merkt op dat L_1 bestaat uit alle woorden in Σ^* waarin maar een soort letter

voorkomt. Dus $L_1 = \{a\}^+ \cup \{b\}^+ \cup \{c\}^+ \cup \{d\}^+$.
Omdat $\{e\}^+$ aftelbaar is voor elk één-letter-alfabet $\{e\}$ (via de bijectie $f(e^n) = n$ voor alle $n \in \mathbb{N}$) is L_1 als eindige vereniging van aftelbare verzamelingen ook weer aftelbaar.
Je geeft expliciet een aftelfunctie: Definieer $g : L_1 \to \mathbb{N}$ door $g(a^n) = 4n - 3$, $g(b^n) = 4n - 2$, $g(c^n) = 4n - 1$ en $g(d^n) = 4n$
voor alle $n \in \mathbb{N}$. Deze functie is een bijectie.

(f) L_1 en Σ^* zijn gelijkmachtig, want beide zijn oneindig en aftelbaar, dus allebei gelijkmachtig met $\mathbb{N}$ (en gelijkmachtig zijn is een symmetrische, transitieve relatie).

Opgave 2

(a) $f(b) = f(b \cdot \lambda) = f(\lambda) \cdot b = \lambda \cdot b = b$
$f(ab) = a \cdot f(b) = a \cdot b = ab$ (zie boven)
$f(ba) = f(a) \cdot b = a \cdot f(\lambda) \cdot b = a \cdot \lambda \cdot b = ab$
$f(bbab) = f(bab) \cdot b = f(ab) \cdot b \cdot b = ab \cdot b \cdot b = abbb$

(b) $f(u) = a^2b$ als in u twee a's en één b voorkomen; d.w.z. $u = aab$ of $u = aba$ of $u = baa$.
De functie f zet altijd a's voorop en b's achteraan, dus er zijn geen u met $f(u) = b^2a$.

(c) f is totaal, want $f(\lambda)$ is gedefinieerd en als $f(x)$ gedefinieerd is dan ook $f(ax)$ en $f(bx)$. Dit is een inductief argument dat laat zien dat f gedefinieerd is voor elk woord in $\{a, b\}^*$.
f is niet injectief, want $f(ab) = f(ba) = ab$ (zie onderdeel a).
f is niet surjectief, want er is geen origineel voor b^2a (zie onderdeel b).

(d) Te bewijzen: voor alle $w \in \{a, b\}^*$ is $f(w) = a^{|w|_a}b^{|w|_b}$.
Bewijs met inductie naar $|w|$:
(B) $|w| = 0$, dus $w = \lambda$. Dan is $f(w) = f(\lambda) = \lambda = a^0b^0$. Omdat $|\lambda|_a = |\lambda|_b = 0$, voldoet $f(w)$ aan de eis.
(IH) Er is een $k \in \mathbb{N}_0$ zodat voor alle $x \in \{a, b\}^*$ met $|x| = k$ geldt dat $f(x) = a^{|x|_a}b^{|x|_b}$.
(IS) Bekijk nu $w \in \{a, b\}^*$ zodat $|w| = k + 1$. Dus $w = ax$ of $w = bx$ voor een $x \in \{a, b\}^*$ met $x = k$.
Als $w = ax$, dan is $|w|_a = |x|_a + 1$ en $|w|_b = |x|_b$. Volgens de definitie is $f(w) = f(ax) = a \cdot f(x)$ wat volgens de inductiehypothese gelijk is aan $a \cdot a^{|x|_a}b^{|x|_b} = a^{|x|_a+1}b^{|x|_b}$. Dus $f(w) = a^{|w|_a}b^{|w|_b}$.
Als $w = bx$, dan is $|w|_a = |x|_a$ en $|w|_b = |x|_b + 1$. Volgens de definitie is $f(w) = f(xb) = f(x) \cdot b$ wat volgens de inductiehypothese gelijk is aan $a^{|x|_a}b^{|x|_b} \cdot b = a^{|x|_a}b^{|x|_b+1}$. Dus $f(w) = a^{|w|_a}b^{|w|_b}$.

Opgave 3

(a) M heeft begintoestand s en eindtoestand f. Dus
$L(M) = \{x \in \{1\}^* : sx \vdash^* g_1 f g_2 \text{ met } g_1, g_2 \in \{1, B, a, \$\}^*\}$
f_M wordt gedefinieerd door: voor alle $x \in \{1\}^*$,
$f_M(x) = y$ dan en slechts dan als $sx \vdash^* y_1 f y_2$ met $y = y_1 y_2$ en $y \in \{1, a, \$\}^*$.

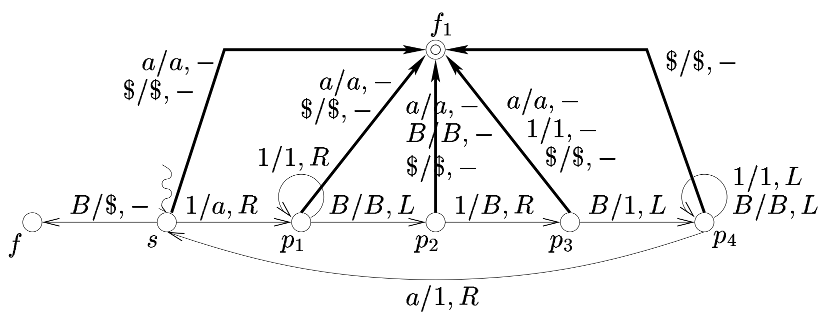
(b) $s \vdash f\$ \qquad$ dus $f_M(\lambda) = \$$
$s11 \vdash ap_1 1 \vdash a1p_1 \vdash ap_2 1 \vdash aBp_3 \vdash ap_4 B1 \vdash p_4 aB1 \vdash 1sB1 \vdash 1f\$1$
dus $f_M(11) = 1\$1$

(c) Het ligt het meest voor de hand om te onderzoeken wat M doet met de kortste overgebleven invoerwoorden:
$s1 \vdash ap_1 \vdash p_2 a \not\vdash$ hangt!
$s111 \vdash ap_1 11 \vdash a1p_1 1 \vdash a11p_1 \vdash a1p_2 1 \vdash a1Bp_3 \vdash a1p_4 B1 \vdash ap_4 1B1 \vdash p_4 a1B1 \vdash 1s1B1 \vdash 1ap_1 B1 \vdash 1p_2 aB1 \not\vdash$ hangt!
Dus noch 1 noch 111 worden geaccepteerd.

Opgave 4

(a) $\mathcal{L}_{DTM} = \{L : L = L(M), M \text{ een DTM}\}$
$\mathcal{L}_{REC} = \{L : L = Y(M), M \text{ een beslisser}\}$.

(b) Omdat M voor geen enkel invoerwoord een oneindige berekening heeft, bestaat het complement van $L(M_1)$ precies uit alle woorden over het invoeralfabet $\{1\}$ waarop de berekening van M gaat hangen. M hangt als er in een toestand een symbool wordt gelezen waarvoor in die toestand geen instructie is.



We passen M als volgt aan:
De oude eindtoestand f wordt een gewone toestand (zonder instructies), dus alle berekeningen die eerst in die toestand succesvol stopten, gaan daar nu hangen.
Er komt een nieuwe eindtoestand f_1. Voor elke combinatie van toestand q en bandsymbool $X \in \{1, B, a, \$\}$ waarvoor M geen instructie heeft, krijgt $\overline{M}$ een instructie $(q, X, f_1, X, -)$. Dus alle berekeningen die hingen in M zullen nu succesvol stoppen in de nieuw toegevoegde eindtoestand f_1.

(c) $L(M) \in \mathcal{L}_{REC}$, omdat zowel $L(M)$ als zijn complement talen uit $\mathcal{L}_{DTM}$ zijn. Er geldt immers dat $\mathcal{L}_{REC} = \{L \in \mathcal{L}_{DTM} : \overline{L} \in \mathcal{L}_{DTM}\}$, waarbij $\overline{L}$ het complement van L is.

Meer constructief en direct, kan je ook opmerken dat uit M ook een beslisser geconstrueerd kan worden met $L(M)$ als zijn ja-taal. Cruciaal is dat de berekeningen van M altijd eindigen. Succesvol stoppen in M leidt in de beslisser tot stoppen met een **y** op de band en hangen in M leidt tot stoppen met een **n** op de band. (Bedenk wel dat de band ook nog leeg moet worden gemaakt op het symbool **y** of **n** na.)

Opgave 5
Een macht van 2 is een even getal, dat gehalveerd weer een macht van 2 is ($2^n = 2^{n-1} + 2^{n-1}$ als $n \geq 1$ en $2^0 = 1$). Je kan dus vaststellen of je met een macht van 2 te doen hebt, door zolang het getal even is te halveren totdat je bij 1 uitkomt. Er zijn verschillende manieren om te halveren. Je kan bijvoorbeeld (van L naar R gaande over de band) voor elke 1 die je tegenkomt, de volgende 1 die je ziet "wegstrepen" (vervangen door een ander symbool). Hieronder wordt een DTM programma gegeven waarin het halveren gebeurt door voor elke 1 aan het begin, een 1 aan het eind weg te strepen.

stap 0. als de invoer leeg is, hangt de Turingmachine meteen ($k = 0$ en 0 is geen macht van 2).
stap 1. als een 1 wordt gelezen en daarna staat gelijk een B, dan is $k = 1 = 2^0$; het woord wordt geaccepteerd, de Turingmachine gaat over in zijn halttoestand.
stap 2. als een 1 wordt gelezen en rechts daarvan staat nog een 1, dan wordt hij (de linkse van die twee 1-en) gemarkeerd als a, de kop gaat helemaal naar rechts en veegt de laatste 1 uit.
stap 3. de kop gaat terug naar links, naar de a.
stap 4. de a wordt weer een 1, de kop gaat één cel naar rechts.
stap 5. als nu een B wordt gelezen, is k gehalveerd,
de kop loopt terug naar links, naar de voorste 1 en we gaan verder met stap 1.
stap 6. als nu een 1 wordt gelezen en
6.1 daarna staat gelijk een B, dan was k oneven, de Turingmachine hangt;
6.2 anders staat daarna een 1 en we zijn weer in stap 2.