TENTAMEN FUNDAMENTELE INFORMATICA 1 (I&E)
Dinsdag 23 december 2014, 14.00 - 17.00 uur

Dit tentamen bestaat uit 8 opgaven, waarbij steeds tussen [en] staat hoeveel punten er ongeveer mee te verdienen zijn. In totaal zijn er 100 punten te verdienen.

Als je het antwoord op een onderdeel niet weet, en je hebt dat antwoord nodig bij een later onderdeel, dan kun je het antwoord 'kopen' bij de docent.

Geef gevraagde eindige automaten en Turing machines door middel van hun transitiediagram (het plaatje dus).

Als er bij een opgave gevraagd wordt om uitleg bij je antwoord, is het belangrijk dat je die ook geeft.

---

1. [10 pt]

   (a) Laat $L_a = \{\Lambda, aa\}$ en $L_b = \{b\}$. Geef de eerste zes elementen in de canonieke volgorde van de taal $L_a(L_a^* \cup L_b)$.

   (b) Geldt er algemeen, voor willekeurige talen L_1 en L_2, dat

   $$L_1^* \cup L_1 L_2 = L_1(L_1^* \cup L_2)$$

   Zo ja, toon dit formeel aan.
   Zo nee, geef een voorbeeld van talen L_1 en L_2 en een string x die wel in de ene resulterende taal zit (dus in $L_1^* \cup L_1 L_2$ of in $L_1(L_1^* \cup L_2)$), en niet in de andere. Zeg er uiteraard ook bij, in welke resulterende taal x dan wel zit en in welke niet.

---

2. [12 pt] Het pomplemma voor reguliere talen luidt als volgt:

   Stel L is een taal over het alfabet Σ.
   Als L geaccepteerd wordt door een eindige automaat $M = (Q, \Sigma, q_0, A, \delta)$,
   en als n het aantal toestanden van M is,
   dan zijn er voor elke $x \in L$ met $|x| \geq n$ drie strings u, v en w
   zó dat $x = uvw$ en de volgende drie beweringen waar zijn:
   1. $|uv| \leq n$.
   2. $|v| > 0$ (dwz: $v \neq \Lambda$).
   3. Voor elke $m \geq 0$ zit ook de string $uv^m w$ in L.

   Gebruik dit pomplemma om aan te tonen dat de taal

   $$L_0 = \{(ab)^i a^j b^k \quad | \quad i, k \geq 0 \text{ en } j > i + k\}$$

   niet door een eindige automaat geaccepteerd kan worden. Inderdaad, L_0 is de taal uit de huiswerkopgave.

   Ofwel: veronderstel dat L_0 wél door een eindige automaat geaccepteerd kan worden, kies dan een geschikt woord $x \in L_0$ en toon aan dat x niet opgepompt en/of weggepompt kan worden. Vergeet ook niet om de conclusie te trekken.

---

3. [11 pt] Laat L de taal zijn van de woorden $x \in \{a, b\}^*$ met de volgende drie eigenschappen (alle drie de eigenschappen):

   - x bevat één substring aa,
   - x bevat geen andere a's,
   - en $n_b(x)$ is een veelvoud van 3, d.w.z.: 0, 3, 6, 9, 12, 15, ...

   (a) Geef een eindige automaat M, zó dat $L(M) = L$. De automaat mag bij geen enkele invoer crashen, dus voeg zo nodig een afvoerputje toe.

   (b) Geef een reguliere expressie die met de taal L overeenkomt. Beredeneer dat je expressie inderdaad precies met L overeenkomt.

---

4. [12 pt]

   (a) Laat M een willekeurige niet-deterministische eindige automaat (NFA) zijn, en laat $L = L(M)$. Leg uit (met woorden, plaatjes en/of formules, maar in ieder geval duidelijk en volledig) hoe je M kunt ombouwen naar een NFA M', zó dat $L(M') = L^*$.

   (b) Geef een NFA die overeenkomt met de reguliere expressie

$$((a + ba)^*a + bba)^*$$

---

5. [22 pt]

   (a) Geef een CFG G_1, zó dat $L(G_1) = L_1$, met

$$L_1 = \{(ab)^i a^j b^k \mid i, j, k \geq 0 \text{ en } i \leq j < i + k\}$$

   Leg uit wat de functie is van de diverse variabelen en producties in G_1.

   (b) Geef een afleiding van het woord *abaabb* in de grammatica G_1 van het vorige onderdeel.

   (c) Geef de eerste zeven elementen in de canonieke volgorde van de taal

$$L_2 = \{(ab)^i a^j b^k \mid i, j, k \geq 0 \text{ en } j < i + k\}$$

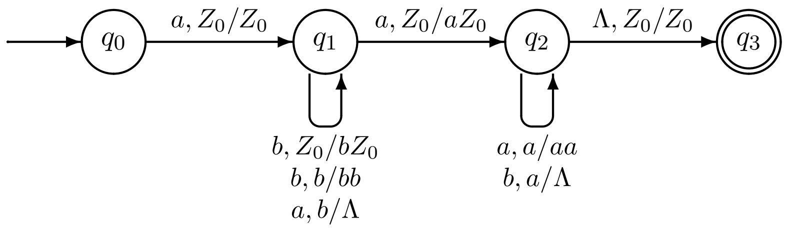
   (we eisen dus niet meer dat $i \leq j$).

   N.B.: het is mogelijk dat een woord x overeenkomt met verschillende combinaties van i, j en k. Als er maar (minstens) één combinatie is die aan de voorwaarde $j < i + k$ voldoet, behoort x tot L_2.

   (d) Geef een CFG G_2, zó dat $L(G_2) = L_2$.

---

6. [10 pt] Beschouw de volgende stapelautomaat M:



Beschrijf (met woorden en/of formules, maar in ieder geval duidelijk en volledig) de strings die geaccepteerd worden door M.

Hint: Bedenk wat voor substrings je kunt doorlopen bij de lusjes op q_1 en q_2, voordat je verder kunt naar de volgende toestand.

---

7. [9 pt]

   (a) Bij een stapelautomaat (PDA) $M = (Q, \Sigma, \Gamma, q_0, Z_0, A, \delta)$ stelt δ de transitiefunctie voor. Leg uit in woorden wat het betekent als voor zekere toestand p, invoerletter σ en stapelsymbool X geldt dat

$$\delta(p, \sigma, X) = \{(q_1, \alpha_1), (q_2, \alpha_2)\}$$

   (b) Aan welke voorwaarde(n) moet de transitiefunctie van een PDA M voldoen, voordat we M deterministisch mogen noemen? (Het is niet voldoende om te zeggen dat je bij het verwerken van een invoerstring door M geen keuze tussen verschillende transities hebt.)

---

8. [14 pt] We noemen een element $x \in \{0,1\}^*$ een *binaire representatie van een natuurlijk getal* als x aan de volgende voorwaarden voldoet:

   - $x \neq \Lambda$,
   - en x heeft geen onnodige voorloopnullen, d.w.z.: als x met een 0 begint, dan moet x gelijk zijn aan de string 0.

   In het algemeen vinden we, uitgaande van een binaire representatie x van een natuurlijk getal, de volgende binaire representatie door de achterste 0 in x te vervangen door een 1, en de eventuele 1'en daarachter te vervangen door een 0. Bijvoorbeeld: 1010111 levert dan 1011000 op. Alleen wanneer x geen 0 bevat, werkt dit niet. In dat geval veranderen we alle 1'en in x in een 0, en komt er een 1 voor te staan. Bijvoorbeeld: 1111 levert dan 10000 op.

   Construeer een Turing machine T die als invoer een string $x \in \{0,1\}^*$ heeft, en daarvoor de volgende partiële functie f berekent:

$$f(x) = \begin{cases} \text{de volgende binaire representatie} & \text{als } x \text{ de binaire representatie} \\ & \qquad\text{van een natuurlijk getal is} \\ \text{ongedefinieerd} & \text{anders} \end{cases}$$

   Leg duidelijk uit hoe T werkt.
