HERTENTAMEN AUTOMATA THEORY
Donderdag 3 februari 2022, 10.15 - 13.15 uur

Dit tentamen bestaat uit acht opgaven, waarbij steeds tussen [en] staat hoeveel punten er ongeveer mee te verdienen zijn. In totaal zijn er 100 punten te verdienen. Wanneer er bij een vraag om uitleg, motivatie of toelichting gevraagd wordt, is het belangrijk om die ook te geven.
Als we het in dit tentamen over een eindige automaat hebben (zonder verdere toevoeging), bedoelen we een deterministische eindige automaat zonder Λ-transities (wat elders *DFA* genoemd wordt).

---

1. [9 pt] Laat

$$L = \{x \in \{a,b\}^* \mid x \text{ bevat minstens een voorkomen van } aa \text{ en een voorkomen van } abb\}$$

Merk op dat voorkomens van substrings kunnen overlappen. Bijvoorbeeld, $aabb \in L$.

Teken een eindige automaat M, zó dat $L(M) = L$.

---

2. [12 pt]

(a) Laat $M_1 = (Q_1, \Sigma, q_1, A_1, \delta_1)$ en $M_2 = (Q_2, \Sigma, q_2, A_2, \delta_2)$ twee willekeurige eindige automaten met hetzelfde invoeralfabet Σ zijn. Laat $L_1 = L(M_1)$ en $L_2 = L(M_2)$. Met de productconstructie kun je uit M_1 en M_2 een eindige automaat $M = (Q, \Sigma, q_0, A, \delta)$ construeren, zó dat $L(M) = L_1 \cup L_2$.

Beschrijf (in woorden en/of formules, maar in ieder geval duidelijk en volledig) wat bij deze automaat M de onderdelen Q, q_0, A en δ zijn.

(b) Leg uit hoe je de productconstructie uit onderdeel (a) moet aanpassen, om de taal

$$\{x \in \Sigma^* \mid \text{als } x \in L_1, \text{ dan is ook } x \in L_2\}$$

te accepteren. Licht ook kort toe waarom de aangepaste constructie correct is.

---

3. [16 pt]

   (a) Laat $L \subseteq \Sigma^*$ een taal zijn, en laat $x \in \Sigma^*$ een string zijn.
   Hoe is de verzameling L/x (de 'toekomstverzameling' van x met betrekking tot L) gedefinieerd? Dat wil zeggen: welke strings zitten er in L/x?
   Als je het antwoord op dit onderdeel niet weet, dan kun je het 'kopen' van de docent. Wellicht kun je dan wel onderdelen (b) en (c) maken.

   Laat nu

$$\begin{aligned} L_1 \;=\; \{x \in \{a,b\}^* \;\mid\; & x = a^i(ba)^j a^k \text{ met } i,j,k \geq 0 \\ & \text{en } x \text{ bevat evenveel voorkomens van de substring } ba \text{ als van de substring } aa\} \end{aligned}$$

   Een string $x \in L_1$ moet dus aan alle vermelde eisen voldoen. Merk op dat voorkomens kunnen overlappen. Bijvoorbeeld $x = aaa$ bevat twee voorkomens van de substring aa.

   Inderdaad, deze taal L_1 kennen we van de huiswerkopgaven. De eerste acht elementen in de canonieke volgorde van L_1 zijn $\Lambda, a, baa, aaba, abaa, babaaa, aaababa, aababaa$.

   (b) Laat $m \geq 0$ willekeurig zijn,[1] en laat $x = a^m$. Wat is, bij bovenstaande taal L_1, de verzameling L_1/x? Onderscheid in je antwoord het geval dat $m = 0$ en het geval dat $m \geq 1$.
   Beschrijf in je antwoord concreet wat de elementen van L_1/x zijn, in termen van a, ba, m, i, j en k. Alleen als de toekomstverzameling de hele taal L_1 zou zijn, mag je gewoon L_1 zeggen.

   (c) Bij een eerder tentamen hebben we gezien dat twee strings a^m en a^n met $m, n \geq 0$ en $m \neq n$ altijd L_1-distinghuishable zijn. Ofwel: dat ze verschillende toekomstverzamelingen hebben.
   Geef nu twee verschillende, concrete strings $x, y \in \{a,b\}^*$, zó dat $L_1/x = L_1/y \neq \emptyset$. Ofwel: twee verschillende strings die wel dezelfde, niet-lege toekomstverzameling hebben. Geef ook deze gemeenschappelijke toekomstverzameling $L_1/x = L_1/y$. Beschrijf in je antwoord concreet wat de elementen van $L_1/x = L_1/y$ zijn, in termen van a, ba, i, j en k.
   N.B.: Uiteraard moet minstens een van de strings x en y (minstens) een voorkomen van de letter b bevatten.

---

[1] Dit betekent niet dat je een willekeurige $m \geq 0$ mag *kiezen*, maar dat m elk niet-negatief getal kan zijn.

4. [10 pt] Laat

$$L = \{x \in \{a,b\}^* \quad | \quad |x| \geq 2 \text{ en } x \text{ begint en eindigt met } aa \text{ of } ba\}$$

Enkele elementen van L zijn bijvoorbeeld aaa en $aababa$. Geef voor elk van de hierna volgende reguliere expressies r_1, r_2, r_3, r_4 aan of het een correcte expressie voor L is of niet. Als een r_i geen correcte expressie is, licht dat dan toe, door een string x te geven

(a) die wel in L zit, maar niet aan r_i voldoet,

(b) of die niet in L zit, maar wel aan r_i voldoet.

Als voor beide situaties strings x te vinden zijn, hoef je toch maar één string x te geven. Maak wel duidelijk of situatie (a) of (b) van toepassing is voor een string x die je geeft.

$$\begin{array}{ll} r_1: & (a+b)(a+b)^*a \\ r_2: & (a+b)a(a+b)^*a + aa \\ r_3: & (aa+ba)(b^*a)^* \\ r_4: & (aa+ba)((a+b)^*(aa+ba))^* \end{array}$$

---

5. [10 pt] Hieronder geven we twee grammatica's G_1 en G_2, die ooit werden voorgesteld voor de taal L_1 uit opgave 3. De grammatica's zijn niet per se correct voor taal L_1, maar daar gaat het nu niet om. Beantwoord voor elke G_i de volgende vragen:

Is G_i dubbelzinnig?

Zo ja, toon dat aan door twee verschillende afleidingsbomen voor een zelfde string $x \in L(G_i)$ te geven.

Zo nee, toon dat aan door te beredeneren dat met verschillende producties in G_i verschillende strings worden afgeleid. Je hoeft geen compleet inductiebewijs te geven.

(a) G_1 is de context-vrije grammatica met startvariabele S en de volgende producties:

$$\begin{array}{lcl} S & \rightarrow & AB \mid aAB \mid \Lambda \\ A & \rightarrow & aAba \mid \Lambda \\ B & \rightarrow & baBa \mid \Lambda \end{array}$$

(b) G_2 is de context-vrije grammatica met startvariabele S en de volgende producties:

$$\begin{array}{lcl} S & \rightarrow & aA \mid B \mid aaAbabaBa \\ A & \rightarrow & aAba \mid \Lambda \\ B & \rightarrow & baBa \mid \Lambda \end{array}$$

---

6. [15 pt]

   Laat
   $$L = \{a^i b^j c^k \quad | \quad i, j, k \geq 0 \text{ en } i < j + k\}$$

   (a) Geef de eerste zes elementen in de canonieke volgorde van L.

   (b) Geef een context-vrije grammatica G, zó dat $L(G) = L$.

   Leg ook uit wat de functie is van de diverse variabelen en producties in G voor het genereren van L.

---

7. [14 pt]

   Laat
   $$L = \{a^i b^j c^k \quad | \quad i, j, k \geq 0 \text{ en } i \neq j + k\}$$

   N.B.: deze taal is net iets anders dan de taal L in opgave 6. Nu mag i ook groter dan $j + k$ zijn.

   Teken een stapelautomaat M, zó dat $L(G) = L$. Probeer ervoor te zorgen dat M deterministisch is. en geen Λ-transities bevat. Als dit niet lukt, kun je nog wel het grootste deel van de punten verdienen.

   Leg ook uit hoe M zijn toestanden en stapel gebruikt om precies de juiste taal te accepteren.

---

8. [14 pt] Laat G de context-vrije grammatica zijn met startvariabele S en de volgende producties:

   $$S \to aSb \mid T \qquad T \to aT \mid \Lambda$$

   (a) Teken de niet-deterministische top-down stapelautomaat $NT(G)$.

   Als je het antwoord op dit onderdeel niet weet, dan kun je het 'kopen' van de docent. Wellicht kun je dan wel onderdelen (b) en (c) maken.

   (b) Voer een succesvolle berekening uit in $NT(G)$ voor de invoer $x = aab$, d.w.z.: een berekening die begint in de initiële configuratie voor x en resulteert in acceptatie van x.

   Geef deze berekening net als bij de colleges weer met een tabel met kolommen voor (1) de toestand, (2) de reeds gematchte invoer, (3) de resterende invoer, (4) de stapelinhoud, en (5) de expand- of match-actie (indien van toepassing).

   (c) Geef de links-preferente afleiding van aab in G die overeenkomt met de berekening uit onderdeel (b).

`einde tentamen`