HERTENTAMEN COMPUTABILITY
Donderdag 2 juni 2022, 10.15 - 13.15 uur

Dit tentamen bestaat uit vijf 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.

---

1. [25 pt]

   (a) Teken een gewone (deterministische, 1-tape) Turingmachine T_1, zó dat

   $$L(T_1) = XX = \{ss \mid s \in \{a,b\}^*\}$$

   Zorg ervoor dat T_1 voor invoerstrings x die niet in XX zitten, eindigt in h_r, en teken ook de daarvoor benodigde transities naar h_r.
   Als je voor T_1 gebruik wilt maken van componenten, zul je die componenten ook moeten uitwerken (tekenen dus).
   Leg ook duidelijk uit hoe T_1 werkt.

   (b) Laat

   $$L_2 = \{sst \mid s,t \in \{a,b\}^* \text{ en } |s| \geq 1\}$$

   Dus bijvoorbeeld $abab \in L_2$ (met $t = \Lambda$), $ababaa \in L_2$ (met $t = aa$) en $aabaa \in L_2$ (met $t = baa$).
   Teken een *niet-deterministische* Turingmachine T_2, zó dat $L(T_2) = L_2$.
   Je mag voor T_2 gebruik maken van T_1 (uit onderdeel (a)) als component. Als je T_1 daarbij iets moet aanpassen, beschrijf de benodigde aanpassing dan precies. Andere componenten mag je alleen gebruiken als je ze zelf ook tekent.

---

2. [18 pt] Bij een eerder tentamen werd gevraagd om een unrestricted grammar voor de taal

$$L = \{a^i b^i c^j \ \mid \ i, j \geq 0 \text{ en } i \neq j\}$$

Wellicht ten overvloede: de eerste acht elementen in de canonieke volgorde van L zijn: $c, ab, cc, ccc, aabb, abcc, cccc, aabbc$.

Hieronder moet je vier keer een vraag met 'ja' of 'nee' beantwoorden. Als het antwoord 'ja' is, hoef je dat niet toe te lichten.

(a) Laat G_1 de unrestricted grammar zijn met startvariabele S en de volgende producties:

$$\begin{array}{rcl} S & \rightarrow & LIR \mid LJR \\ I & \rightarrow & ABIC \mid ABI \mid AB \\ J & \rightarrow & ABJC \mid JC \mid C \\ BA & \rightarrow & AB \end{array}$$

$$LA \rightarrow a \qquad aA \rightarrow aa \qquad aB \rightarrow ab \qquad bB \rightarrow bb \qquad CR \rightarrow c \qquad Cc \rightarrow c$$

Hierin is variabele I bedoeld voor het geval $i > j$, en J voor het geval $j > i$.

i. Is $L(G_1) \subseteq L$? Zo nee, geef een element x van $L(G_1)$ dat niet in L zit. Geef in dat geval ook een afleiding van x in G_1. Als je in die afleiding meerdere 'gelijksoortige' producties achter elkaar moet toepassen, mag je die stappen samenvatten met $\Rightarrow^*$.

ii. Is $L \subseteq L(G_1)$? Zo nee, geef een element x van L dat niet in $L(G_1)$ zit. Beredeneer in dat geval ook waarom x niet afgeleid kan worden in G_1.

(b) Laat G_2 de unrestricted grammar zijn met startvariabele S en de volgende producties:

$$\begin{array}{rcl} S & \rightarrow & aBI \mid Jc \\ I & \rightarrow & aBaBIc \mid aBI \mid \Lambda \\ J & \rightarrow & aBJcc \mid Jc \mid \Lambda \\ Ba & \rightarrow & aB \end{array}$$

$$aB \rightarrow ab \qquad bB \rightarrow bb$$

Ook hierin is variabele I bedoeld voor het geval $i > j$, en J voor het geval $j > i$.

i. Is $L(G_2) \subseteq L$? Zo nee, geef een element x van $L(G_2)$ dat niet in L zit. Geef in dat geval ook een afleiding van x in G_2. Als je in die afleiding meerdere 'gelijksoortige' producties achter elkaar moet toepassen, mag je die stappen samenvatten met $\Rightarrow^*$.

ii. Is $L \subseteq L(G_2)$? Zo nee, geef een element x van L dat niet in $L(G_2)$ zit. Beredeneer in dat geval ook waarom x niet afgeleid kan worden in G_2.

3. [19 pt] In deze opgave mag je ervan uitgaan dat $\Sigma = \{a, b\}$.

   (a) Laat $T = (Q, \Sigma, \Gamma, q_0, \delta)$ een Turingmachine zijn. Geef de initiële configuratie van T voor de invoer $x = ab$.

   Als je het antwoord op dit onderdeel niet weet, dan kun je het 'kopen' van de docent. Wellicht heb je er iets aan voor onderdeel (c).

   In het bewijs van Stelling 8.14 in het boek wordt beschreven, hoe je bij een willekeurige Turingmachine $T = (Q, \Sigma, \Gamma, q_0, \delta)$ een unrestricted grammar $G = (V, \Sigma, S, P)$ kunt construeren, zó dat $L(G) = L(T)$.

   De grammatica G bevat drie soorten producties. De eerste soort is bedoeld om initiële configuraties van de Turingmachine na te bouwen, met twee kopies van de tape-inhoud. De tweede soort is voor het simuleren van berekeningen van de Turingmachine.

   (b) Wat is het doel van de producties van de derde soort?

   (c) Geef de producties van de eerste soort in G (of, althans, producties die hetzelfde effect hebben).

   (d) Geef alle producties van de tweede soort in G die volgens de constructie van Stelling 8.14 corresponderen met de volgende transitie van een Turingmachine:

   $$p_1 \xrightarrow{a/A,\mathrm{R}} q_1$$

   Wanneer je in je beschrijving van de producties variabele(n) σ_i gebruikt, zeg dan wel wat de waardes van de σ_i('s) kan/kunnen zijn. Als alternatief mag je ook gebruiken dat $\Sigma = \{a, b\}$ en $\Gamma = \{a, b, A, B\}$.

   (e) Geef alle producties van de tweede soort in G die volgens de constructie van Stelling 8.14 corresponderen met de volgende transitie van een Turingmachine:

   $$p_2 \xrightarrow{b/B,\mathrm{L}} q_2$$

   Wanneer je in je beschrijving van de producties variabele(n) σ_i gebruikt, zeg dan wel wat de waardes van de σ_i('s) kan/kunnen zijn. Als alternatief mag je ook gebruiken dat $\Sigma = \{a, b\}$ en $\Gamma = \{a, b, A, B\}$.

Z.O.Z.

4. [12 pt]

   (a) Laat $L \subseteq \Sigma^*$ een taal zijn. Hoe is de karakteristieke functie χ_L gedefinieerd?

   Als je het antwoord op dit onderdeel niet weet, dan kun je het 'kopen' bij de docent. Wellicht kun je dan wel het hierna volgende onderdeel maken.

   (b) Toon aan dat als er een Turingmachine T_1 bestaat die L accepteert en die halt voor elke mogelijke invoer x, dat er dan ook een Turingmachine T_2 bestaat die de karakteristieke functie χ_L berekent.

   Doe dit door de werking van zo'n Turingmachine T_2 te beschrijven (op het niveau van de functionaliteit). Behandel daarbij ook wat nodig is om ervoor te zorgen dat T_2 inderdaad de juiste uitvoer op de tape kan achterlaten.

---

5. [26 pt] Beschouw het volgende beslissingsprobleem:

   WritesNonblank: Gegeven een Turingmachine T_1, schrijft T_1 bij invoer Λ ooit een nonblank symbol op de tape?

   (a) Toon aan dat het beslissingsprobleem *WritesNonblank* beslisbaar is. Doe dit door een algoritme te beschrijven dat het probleem beslist, en door aan te tonen dat dit algoritme correct is.

   (b) Beschouw nu ook het volgende beslissingsprobleem

   Accepts-Λ: Gegeven een Turingmachine T_2, is $\Lambda \in L(T_2)$?

   Toon aan dat *WritesNonblank* $\leq$ *Accepts*-Λ.

   Beschrijf hierbij precies hoe je een instantie van het ene probleem ombouwt tot een instantie van het andere probleem. Het is hierbij niet de bedoeling dat je het algoritme van onderdeel (a) gebruikt. Laat uiteraard ook zien dat aan alle eisen van een reductie is voldaan.

`einde tentamen`