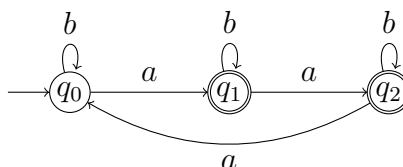


Hertentamen Automaten & Complexiteit

Vrije Universiteit, 11 juni 2014, 18:30–21:15

Opmerkingen. Bij dit tentamen mogen kopieën van de slides worden gebruikt, zonder handgeschreven aantekeningen. Het tekstboek van Linz, laptop en rekenmachine mogen **niet** worden gebruikt! Dit tentamen bestaat uit 9 opgaven verdeeld over 2 pagina's. In totaal zijn 90 punten te behalen; iedere student krijgt 10 punten bonus. Succes!

1. (6 ptn) Zij L de taal van alle strings over $\{a, b\}$ die *geen* substring *abba* bevatten. Geef een minimale dfa die de taal L accepteert.
2. (10 ptn) Construeer uit onderstaande dfa een reguliere expressie die dezelfde taal beschrijft. Geef alle tussenliggende stappen in deze constructie (d.w.z., geef de generaliseerde transitiegrafen).



3. (12 ptn) Ga met behulp van het string matching algoritme na of *aabaabbba* een substring bevat die in $L(b^*(ba^* + ab^*)(ab)^*b)$ zit.
(Beschrijf de gehele constructie: de bijbehorende nfa, en de on-the-fly constructie van de bijbehorende dfa. Indien er zo'n substring is, geef deze substring!)
4. (10 ptn) Teken een npda die de taal $\{a^i b^j c^j d^j \mid i, j \geq 0\}$ accepteert.
5. (12 ptn) Toon aan met behulp van de pompstelling dat $\{a^i b^j c^i d^j \mid i, j \geq 0\}$ geen contextvrije taal is.

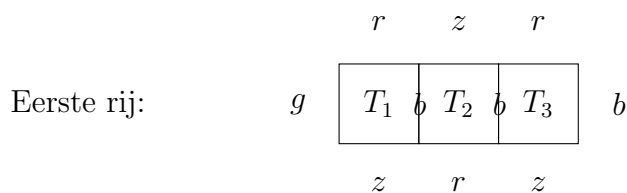
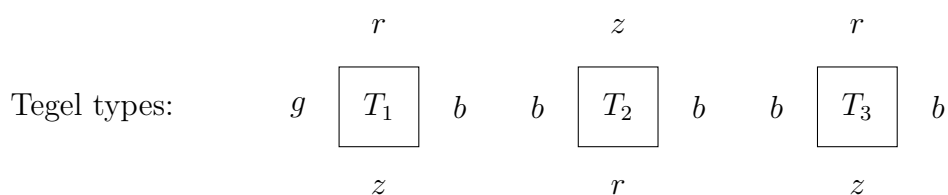
6. (12 ptn) Zij gegeven de volgende grammatica G in Chomsky normaalvorm:

$$\begin{aligned} S &\rightarrow AT \mid BT \mid AB \mid BA \\ T &\rightarrow CS \mid CA \\ A &\rightarrow a \\ B &\rightarrow b \\ C &\rightarrow c \end{aligned}$$

Bereken met behulp van het CYK algoritme of $acab$ in $L(G)$ zit.
(Geef alle tussenstappen.)

7. (12 ptn) Reduceer de volgende instantie van het bounded tiling probleem naar het satisfiability probleem.

(Gebruik $T_1, \dots, T_3$ voor de tegeltypes, en geef de cel linksonder coördinaat $(0, 0)$.)



8. (6 ptn) Is de volgende stelling waar of niet waar? Motiveer je antwoord.

Er zijn problemen die exponentieel veel tijd kosten op een Turing machine, waarvoor een quantum algoritme slechts een polynomiale hoeveelheid tijd nodig heeft.

9. (10 ptn) Beschouw het 2-qubit $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. Stel dat we eerst op het tweede qubit een rotatie van $\frac{\pi}{8}$ toepassen, en dan op het eerste qubit een rotatie van $-\frac{\pi}{8}$. Geef de resulterende superpositie van het 2-qubit in de vorm $\frac{1}{\sqrt{2}}(\alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle)$.
(Geef ook de berekeningen die tot dit antwoord leiden.)