Berekenbaarheid 2014
Uitwerkingen Toets 2
15 december 2014

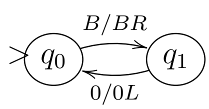
1. Geef een code $R(M)$ van een Turing machine M met (2 punten)

$$R(M) \notin L(M)$$

Verklaar je antwoord.

[Zie de achterkant van dit blaadje voor een relevant stukje uit het boek van Sudkamp.]

Zo'n machine is de machine M_1:



Deze machine heeft code

$$R(M_1) = 000101110110111011001101010101000$$

Omdat M_1 niet termineert met een input die met een nul begint (hij blijft dan oneindig tussen vakje nul en één heen en weer gaan), geldt dus dat $R(M_1)$ niet in de taal van M_1 zit, want M_1 accepteert door stoppen.

2. Het probleem P_2 vraagt bij een gegeven code $R(M)$ of: (2 punten)

$$R(M) \notin L(M)?$$

Gebruik reductie om te bewijzen dat dit probleem onbeslisbaar is.

[Hint: beschouw het probleem $\overline{P_2}$ waarbij ten opzichte van P_2 de antwoorden 'ja' en 'nee' zijn omgewisseld.]

We laten zien dat het blank tape probleem B reduceert naar het probleem $\overline{P_2}$. Hiertoe moeten we een constructie geven die bij iedere machine M een machine M' maakt zodanig dat

$$M(\lambda)\downarrow \iff R(M') \in L(M')$$

Een constructie die deze equivalentie waar maakt is om voor M' te nemen:

- wis tape
- doe M

Als $M(\lambda)$ termineert, termineert M' met iedere input, en dus ook met input $R(M')$. Als $M(\lambda)$ niet termineert, termineert M' met geen enkele input, en dus ook niet met input $R(M')$.

Hieruit volgt dat $\overline{P_2}$ onbeslisbaar is, maar dan is natuurlijk ook P_2 onbeslisbaar.

3. Voldoet het probleem P_2 uit de vorige opgave aan alle voorwaarden van de stelling van Rice? Verklaar je antwoord. ($1\frac{1}{2}$ punt)

   Nee, dit voldoet niet aan de eis dat het probleem naar de eigenschap van een taal vraagt. Er zijn best twee machines te maken die dezelfde taal herkennen, maar waarvan de codes zodanig verschillen dat de ene wel in die taal zit en de andere niet.

4. De taal L_4 is gedefinieerd als ($1\frac{1}{2}$ punt)

$$L_4 := \{R(M) \mid R(M) \notin L(M)\}$$

   Is deze taal recursief? Verklaar je antwoord.

   Nee, deze taal is niet recursief.

   Als deze taal recursief zou zijn, dan was er een Turing machine die deze taal zou herkennen en die voor iedere input stopt (dat is de definitie van een recursieve taal). Maar dat zou dan die machine ook het probleem P_2 beslissen, en we hebben in opgave 2 gezien dat dit probleem onbeslisbaar is.

5. En is de taal L_4 uit de vorige opgave recursief opsombaar? Verklaar je antwoord. (2 punten)

   [Hint: laat eerst met behulp van de universele Turing machine U zien dat het complement van L_4 recursief opsombaar is. Hou er rekening mee dat dit complement ook alle woorden bevat die geen code zijn van een machine.]

   Nee, deze taal is niet recursief opsombaar.

   Het complement van L_4 is recursief opsombaar, want deze wordt herkend door een machine M_5:

- als de input geen code $R(M)$ van een Turing machine is, stop
- verdubbel de input tot $R(M)R(M)$
- doe U

Dit herkent dus precies:

$$\overline{L_4} = \{w \in \{0,1\}^* \mid w \text{ is geen code}\} \cup \{R(M) \mid R(M) \in L(M)\}$$

In één van de opgaven van het werkcollege hebben we geleerd dat een taal recursief is als hij recursief opsombaar is en als zijn complement ook recursief opsombaar is. Dus als L_4 óók recursief opsombaar zou zijn, dan was hij recursief. Maar we hebben in de vorige opgave gezien dat dit niet het geval is. Dus is L_4 niet recursief opsombaar.