Berekenbaarheid 2017
Uitwerkingen Inhaaltoets
27 oktober 2017

1. Laat zien dat het blank tape probleem ook onbeslisbaar is voor Turing-machines met $\Sigma = \{1\}$ en $\Gamma = \{B, 1\}$. (3 punten)

   (Je kunt als input voor dit probleem de code $R(M)$ uit het boek gebruiken waarin geen enkele transitie een 0 leest of schrijft.)

   Noem dit probleem B_1.

   We laten zien dat B_1 onbeslisbaar is door het normale blank tape probleem B (met $\Sigma = \{0, 1\}$) te reduceren naar B_1. Hiertoe moeten we een constructie geven die bij een machine M een machine M' maakt die dezelfde taal herkent maar geen 0 leest of schrijft.

   Een constructie die dit doet is door eerst een tweesporen machine M'' met $\Sigma = \{1\}$ en $\Gamma = \{B, 1\}$ te maken die dezelfde taal herkent, door M te simuleren met een 1 op het onderste spoor als in de oorspronkelijke machine een 0 werd geschreven en met een 1 op het bovenste spoor als in de oorspronkelijke machine een 1 werd geschreven.

   M' is dan de equivalente standaard Turing-machine die met M'' correspondeert. Zoals we die op het college hebben geconstrueerd heeft die dezelfde alfabetten, en voldoet dus aan de gevraagde eigenschap.

2. (a) De taal van het halting probleem is: (2 punten)

   $$L_H = \{R(M)w \mid M(w){\downarrow}\}$$

   Laat zien dat het complement $\overline{L_H}$ van deze taal niet recursief opsombaar is.

   $L_H = L(U)$ is recursief opsombaar. Volgens opgave 26 van hoofdstuk 8 (het werkcollege van 15 september) is een taal recursief dan en slechts dan deze en zijn complement allebei recursief opsombaar zijn. Dus als $\overline{L_H}$ ook recursief opsombaar zou zijn dan zou L_H recursief zijn.

   Maar als L_H recursief is, dan is het halting probleem beslisbaar, en dat is niet zo.

   Dus is L_H niet recursief en dus $\overline{L_H}$ ook niet recursief opsombaar.

   (b) Laat zien dat de taal (2 punten)

   $$L_2 := \{0w \mid w \notin L_H\} \cup \{1w \mid w \in L_H\}$$

   niet recursief opsombaar is, en dat het complement $\overline{L_2}$ van deze taal ook niet recursief opsombaar is.

   Als L_2 recursief opsombaar is, dan wordt hij herkend door een machine M_2. Maar dan wordt $\overline{L_H}$ herkend door de machine M_2' die eerst een 0 voor zijn input plakt, en dan M_2 uitvoert. Maar dan zou $\overline{L_H}$ dus recursief opsombaar zijn, en volgens de vorige deelopgave is dat niet zo. Dus is L_2 ook niet recursief opsombaar..

Net zo wordt $\overline{L_H}$ herkend door een machine die $\overline{L_2}$ herkent, door voor iedere input eerst een 1 te plakken.

3. Zij gegeven een primitief recursieve functie $o : \mathbb{N} \to \mathbb{N}$. Laat zien dat er een partiële μ-recursieve functie $f : \mathbb{N} \to \mathbb{N}$ bestaat zodat het domein van f gelijk is aan het beeld van o, ofwel: (2 punten)

$$\{x \in \mathbb{N} \mid f(x)\downarrow\} = \{x \in \mathbb{N} \mid \exists n \in \mathbb{N}.\, x = o(n)\}$$

(Het omgekeerde geldt ook, al hoort dat niet bij deze opgave: voor iedere μ-recursieve f met niet-leeg domein bestaat een primitief recursieve opsomming o waarvoor deze relatie geldt. Dit verklaart de term 'recursief opsombaar' voor een recursief opsombare taal.)

Een functie met deze eigenschap kan worden gedefinieerd door:

$$f(x) := \mu n.\, \mathsf{eq}(x, o(n))$$