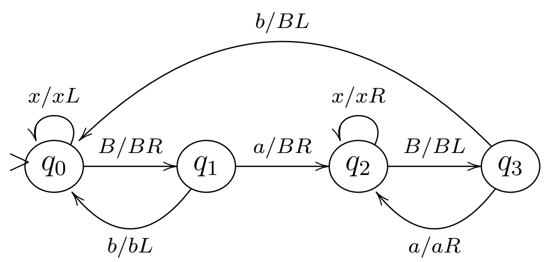
Berekenbaarheid 2013
Uitwerkingen Tentamen
23 januari 2014

1. Geef een standaard Turing-machine die de taal

$$L_1 := \{a^n b^n \mid n \in \mathbb{N}\} = \{\lambda, ab, aabb, \dots\}$$

herkent *door stoppen*. Je mag in je machine hulpsymbolen gebruiken. (Hint: maak eerst de machine die deze taal herkent door eindtoestand, en pas deze vervolgens aan. Je mag deze versie van de machine ook bij de uitwerking erbij geven, maar dat hoeft niet.)

$x \in \{a, b\}$

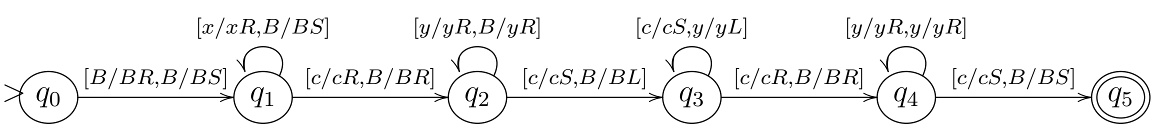


2. Geef een non-deterministische 2-tape Turing-machine die de taal

$$L_2 := \{ucvcvcu' \mid u, u' \in \{a, b, c\}^*, v \in \{a, b\}^*\}$$

herkent door eindtoestand. Zo geldt bijv. dat $abccbcbc \in L_2$, met $u = abc$, $v = b$ en $u' = \lambda$. Zorg er voor dat een correcte input met lengte n wordt herkend in ten hoogste $\frac{3}{2}n + \frac{1}{2}$ stappen.

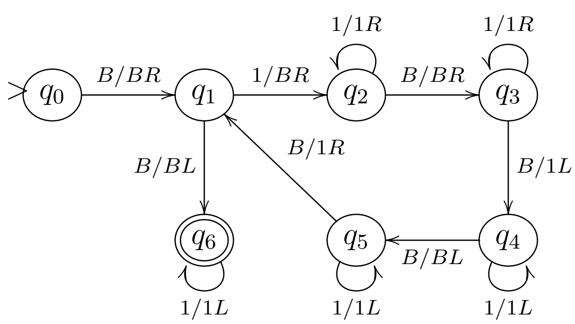
$x \in \{a, b, c\}$, $y \in \{a, b\}$



Het aantal stappen bij een correcte input $ucvcvcu'$ is gelijk aan $1+|u|+1+|v|+1+|v|+1+|v|+1 = |u|+3|v|+5$. Er geldt dat de lengte van deze input gelijk is aan $n = |u|+2|v|+|u'|+3$, en daarom kunnen we rekenen

$\frac{3}{2}n+\frac{1}{2} = \frac{3}{2}|u|+\frac{6}{2}|v|+\frac{3}{2}|u'|+\frac{9}{2}+\frac{1}{2} = \frac{3}{2}|u|+3|v|+\frac{3}{2}|u'|+5 \geq |u|+3|v|+5.$ Dus de grens uit de opgave geldt. (Deze grens is scherp precies dan als $u = u' = \lambda$.)

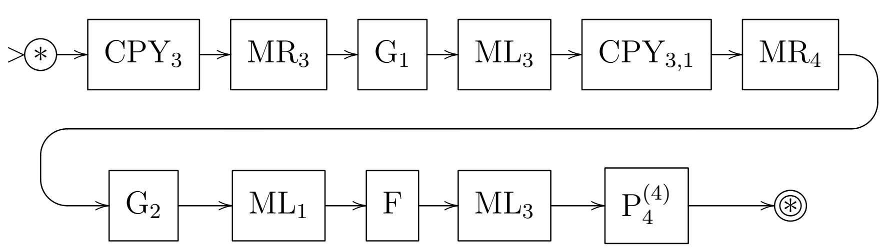
3. Geef een Turing-machine die de macro $\to\boxed{\mathrm{CPY}_1}\to$ implementeert, zoals beschreven op pagina 7. Je mag in deze machine hulpsymbolen gebruiken. Je moet de machine tot op toestanden uitwerken, dus je mag er geen andere macro's in gebruiken.



4. Gegeven macro's $\to\boxed{\mathrm{F}}\to$, $\to\boxed{\mathrm{G}_1}\to$ en $\to\boxed{\mathrm{G}_2}\to$ die de functies $f : \mathbb{N}^2 \to \mathbb{N}$, $g_1 : \mathbb{N}^3 \to \mathbb{N}$ en $g_2 : \mathbb{N}^3 \to \mathbb{N}$ implementeren. Definieer met behulp van deze macro's een macro die de functie

$$f \circ (g_1, g_2)$$

uitrekent. Hou er rekening mee dat je niet weet hoeveel tape deze macro's nodig zullen hebben. Je mag in je machine ook de macro's op pagina 7 gebruiken.

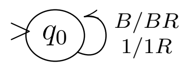


5. Geef een input van de universele Turing-machine U waarbij deze een Turing-machine simuleert die door stoppen bepaalt of

$$101010 \in \{w \in \{0,1\}^* \mid w \text{ bevat een } 0\}$$

Zie pagina 8 voor een relevant stukje uit het boek van Sudkamp.

Een machine die deze taal herkent door stoppen is de machine

M_5 : 

Een code van deze machine is

$$R(M_5) = 0001011101011101100101101011011000$$

De input waarvoor deze machine gesimuleerd moet worden is

$$w = 101010$$

De input voor U is daarom

$$R(M_5)w = 0001011101011101100101101011011000101010$$

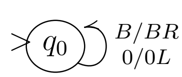
6. Laat zien dat het probleem P_6 onbeslisbaar is dat vraagt of een gegeven Turing-machine M de eigenschap heeft dat er zowel geldt dat hij stopt bij input 1, als dat hij niet stopt bij input 0.

De eigenschap waar dit probleem naar vraagt is equivalent aan

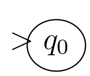
$$1 \in L(M) \text{ en } 0 \notin L(M)$$

Dit is een eigenschap van de taal die herkend wordt door de machine. De onbeslisbaarheid van dit probleem volgt dus met de stelling van Rice als we laten zien dat deze eigenschap niet-triviaal is.

Deze eigenschap geldt wel voor de machine

M_{ja} : 

want deze machine stopt met als input het woord 1, maar niet met input 0. De eigenschap geldt niet voor de machine

M_{nee} : 

want deze machine stopt wél met input 0. Omdat er dus zowel machines bestaan waarvoor de eigenschap geldt als machines waarvoor de eigenschap niet geldt, is deze niet-triviaal.

7. Laat zien dat het probleem P_7 onbeslisbaar is dat vraagt of een gegeven Turing-machine M bij input 0 stopt met output 1, ofwel of $M(0) = 1$.

   We bewijzen onbeslisbaarheid door het blank tape probleem P naar dit probleem te reduceren.

   Dus stel dat we een machine M hebben waarvan we willen weten of hij stopt met als input de blanco tape, ofwel of $M(\lambda)\downarrow$. Er is een Turing machien RED die de code van deze machine omzet naar de code van een machine M' zodat

$$M \text{ voldoet aan de eigenschap van } B \iff M' \text{ voldoet aan de eigenschap van } P_7$$

   wat neerkomt op

$$M(\lambda)\downarrow \iff M'(0) = 1$$

   Een constructie van M' waarvoor deze equivalentie geldt is door M' uit de volgende stappen te laten bestaan:

   (a) wis de tape
   (b) M
   (c) wis de output van $M(\lambda)$
   (d) zet 1 op de tape.

   Als M niet stopt met als input de lege tape dan stopt M' ook niet bij stap (b), en geldt dus $M'(0)\uparrow$ en niet $M'(0) = 1$. Als M wél stopt met als input de lege tape, dan stopt de berekening van stap (b) wel, en geldt dus $M'(0) = 1$ wel.

   (Een subtiliteit is hoe M' weet welk stuk van de tape te wissen in stap (c). Dit punt mag je in de uitwerking van deze opgave negeren, maar kan wel worden opgelost door eerst een machine M'' met hulpsymbolen te maken die bijhoudt waar op de tape is geschreven, en vervolgens deze machine om te zetten in de machine M' met tape alfabet $\{B, 0, 1\}$.)

   Gegeven een machine die P_7 beslist kunnen we dus een machine die B beslist maken door eerst de code $R(M)$ met de machine RED om te zetten in $R(M')$, en vervolgens deze code aan de beslisser van P_7 te geven.

Maar het blank tape probleem B is onbeslisbaar, en dus is er ook geen beslisser voor P_7, en is dit probleem dus ook onbeslisbaar.

8. Geef totale numerieke functies f_8 en g_8 met $f_8 \circ g_8 = \mathsf{id}$ en $g_8 \circ f_8 \neq \mathsf{id}$. Verklaar je antwoord. Geef vervolgens ook de ariteiten van f_8 en g_8.

   Een oplossing is:

$$f_8 = \mathsf{pred}$$
$$g_8 = s$$

   We hebben

$$(f_8 \circ g_8)(x) = f_8(g_8(x)) = \mathsf{pred}(s(x)) = (x+1) \dot{-} 1 = x$$

   dus

$$f_8 \circ g_8 = \mathsf{id}$$

   Evenwel geldt

$$(g_8 \circ f_8)(0) = g_8(f_8(0)) = s(\mathsf{pred}(0)) = (0 \dot{-} 1) + 1 = 0 + 1 = 1 \neq 0 = \mathsf{id}(0)$$

   dus

$$g_8 \circ f_8 \neq \mathsf{id}$$

   De ariteit van f_8 en g_8 zijn allebei gelijk aan 1.

   Een alternative oplossing is:

$$f_8(x) = \lfloor x/2 \rfloor$$
$$g_8(x) = 2x$$

9. Er zijn numerieke functies g en h met

$$\mathsf{mult} = \textbf{primrec}(g, h)$$

   Geef eerst de recursievergelijkingen voor mult. Geef vervolgens de ariteiten van g en h, en schrijf ze als compositie van functies uit het lijstje op pagina 8.

De recursievergelijkingen voor mult zijn:

$$\begin{aligned}\mathsf{mult}(x,0) &= 0\\ \mathsf{mult}(x,y+1) &= \mathsf{mult}(x,y)+x\end{aligned}$$

Dit correspondeert met de functies:

$$\begin{aligned}g(x) &= 0\\ h(x,y,w) &= w+x\end{aligned}$$

Als compositie kunnen we deze functies schrijven als:

$$\begin{aligned}g &= c_0^{(1)}\\ h &= \mathsf{add}\circ(p_3^{(3)},p_1^{(3)})\end{aligned}$$

Hierin heeft g ariteit 1, en h ariteit 3.

10. De numerieke functie ilog is gedefinieerd door

$$\mathsf{ilog}(x,y) = \begin{cases}\lfloor {}^{x}\log(y)\rfloor & \text{als } x \geq 2\\ 0 & \text{als } x \leq 1\end{cases}$$

waarin ${}^{x}\log$ de logaritme met basis x is. We hebben bijv. $\mathsf{ilog}(2,1000) = 9$, want ${}^{2}\log(1000) = 9,96578\ldots$. Dit correspondeert met $2^9 \leq 1000 < 2^{10}$.

Laat zien dat ilog primitief recursief is. Je mag gebruiken dat de functies uit de lijst op pagina 8 alle primitief recursief zijn.

We kunnen ilog schrijven als

$$\mathsf{ilog}(x,y) = \mathsf{ge}(x,2)\cdot \mu n \leq y.\, \mathsf{gt}(y,x^{n+1})$$

of als

$$\mathsf{ilog}(x,y) = \sum_{n=0}^{y} \mathsf{le}(x^n,y)\cdot \mathsf{lt}(y,x^{n+1})\cdot n$$

De vorm van deze schrijfwijzen (expressies met functies uit de lijst op pagina 8 en met begrensde operatoren) impliceert dat de functie primitief recursief is.

Macro's voor Turing-machines voor numerieke berekeningen

$\to\boxed{\mathrm{S}}\to$	$\underset{\uparrow}{B}\,\overline{n}\quad BB$	$\underset{\uparrow}{B}\,\overline{n+1}\,B$
$\to\boxed{\mathrm{Z}}\to$	$\underset{\uparrow}{B}\,\overline{n}\quad\; B$	$\underset{\uparrow}{B}\,\overline{0}\ldots B$
$\to\boxed{\mathrm{E}}\to$	$\underset{\uparrow}{B}\,\overline{n}\,B$	*termineert niet*
$\to\boxed{\mathrm{P}_i^{(k)}}\to$	$\underset{\uparrow}{B}\,\overline{n_1}\,B\ldots B\,\overline{n_k}\,B$	$\underset{\uparrow}{B}\,\overline{n_i}\;\,B\ldots\quad\ldots B$
$\to\boxed{\mathrm{C}_m^{(k)}}\to$	$\underset{\uparrow}{B}\,\overline{n_1}\,B\ldots B\,\overline{n_k}\,B$	$\underset{\uparrow}{B}\,\overline{m}\;\,B\ldots\quad\ldots B$
$\to\boxed{\mathrm{A}}\to$	$\underset{\uparrow}{B}\,\overline{n}\,B\,\overline{m}\quad B$	$\underset{\uparrow}{B}\,\overline{n+m}\,BB$
$\to\boxed{\mathrm{D}}\to$	$\underset{\uparrow}{B}\,\overline{n}\qquad\quad B$	$\underset{\uparrow}{B}\,\overline{n \dot{-} 1}\ldots B$
$\to\boxed{\mathrm{SUB}}\to$	$\underset{\uparrow}{B}\,\overline{n}\,B\,\overline{m}\quad\; B$	$\underset{\uparrow}{B}\,\overline{n \dot{-} m}\ldots B$
$\to\boxed{\mathrm{MULT}}\to$	$\underset{\uparrow}{B}\,\overline{n}\,B\,\overline{m}\,B\ldots$	$\underset{\uparrow}{B}\,\overline{n\cdot m}\,B\ldots$
$\to\boxed{\mathrm{MR}_k}\to$	$\underset{\uparrow}{B}\,\overline{n_1}\,B\ldots B\,\overline{n_k}\,B$	$B\,\overline{n_1}\,B\ldots B\,\overline{n_k}\,\underset{\uparrow}{B}$
$\to\boxed{\mathrm{ML}_k}\to$	$B\,\overline{n_1}\,B\ldots B\,\overline{n_k}\,\underset{\uparrow}{B}$	$\underset{\uparrow}{B}\,\overline{n_1}\,B\ldots B\,\overline{n_k}\,B$
$\to\boxed{\mathrm{FR}}\to$	$\underset{\uparrow}{B}B^i\,\overline{n}\,B\qquad (i\geq 0)$	$B^i\underset{\uparrow}{B}\,\overline{n}\,B$
$\to\boxed{\mathrm{FL}}\to$	$B\,\overline{n}\,B^i\underset{\uparrow}{B}\qquad (i\geq 0)$	$\underset{\uparrow}{B}\,\overline{n}\,B^iB$
$\to\boxed{\mathrm{E}_k}\to$	$\underset{\uparrow}{B}\,\overline{n_1}\,B\ldots B\,\overline{n_k}\,B$	$\underset{\uparrow}{B}\ldots\qquad\quad\ldots B$
$\to\boxed{\mathrm{T}}\to$	$\underset{\uparrow}{B}B^i\,\overline{n}\,B\qquad (i\geq 0)$	$\underset{\uparrow}{B}\,\overline{n}\,B^iB$
$\to\boxed{\mathrm{INT}}\to$	$\underset{\uparrow}{B}\,\overline{n}\,B\,\overline{m}\,B$	$\underset{\uparrow}{B}\,\overline{m}\,B\,\overline{n}\,B$
$\to\boxed{\mathrm{CPY}_k}\to$	$\underset{\uparrow}{B}\,\overline{n_1}\,B\ldots B\,\overline{n_k}\,B\ldots\qquad\quad\ldots B$	$\underset{\uparrow}{B}\,\overline{n_1}\,B\ldots B\,\overline{n_k}\,B\,\overline{n_1}\,B\ldots B\,\overline{n_k}\,B$
$\to\boxed{\mathrm{CPY}_{k,i}}\to$	$\underset{\uparrow}{B}\,\overline{n_1}\,B\ldots B\,\overline{n_k}\,B\,\overline{n_{k+1}}\,B\ldots B\,\overline{n_{k+i}}\,B\ldots\qquad\quad\ldots B$	$\underset{\uparrow}{B}\,\overline{n_1}\,B\ldots B\,\overline{n_k}\,B\,\overline{n_{k+1}}\,B\ldots B\,\overline{n_{k+i}}\,B\,\overline{n_1}\,B\ldots B\,\overline{n_k}\,B$
$\to\boxed{\mathrm{BRN}}\xrightarrow{n=0}$, $\downarrow n>0$	$\underset{\uparrow}{B}\,\overline{n}\,B$	$\underset{\uparrow}{B}\,\overline{n}\,B$

Codering van transities

Symbol	Encoding
0	1
1	11
B	111
q_0	1
q_1	11
$\vdots$	$\vdots$
q_n	1^{n+1}
L	1
R	11

Let $en(x)$ denote the encoding of a symbol x. A transition $\delta(q_i, x) = [q_j, y, d]$ is encoded by the string

$$en(q_i)0en(x)0en(q_j)0en(y)0en(d).$$

Primitief recursieve functies

$$\begin{aligned}
\mathsf{id}(x) &= x \\
z(x) &= 0 \\
s(x) &= x+1 \\
p_i^{(k)}(x_1, \dots, x_k) &= x_i \\
c_n^{(k)}(x_1, \dots, x_k) &= n
\end{aligned}$$

$$\begin{aligned}
\mathsf{pred}(y) &= y \dot{-} 1 \\
\mathsf{add}(x,y) &= x + y \\
\mathsf{mult}(x,y) &= x \cdot y \\
\mathsf{sub}(x,y) &= x \dot{-} y \\
\mathsf{exp}(x,y) &= x^y \\
\mathsf{fact}(x) &= x! \\
\mathsf{sg}(x) &= \text{als } x \neq 0 \text{ dan 1 anders 0} \\
\mathsf{cosg}(x) &= \text{als } x \neq 0 \text{ dan 0 anders 1} \\
\mathsf{lt}(x,y) &= \text{als } x < y \text{ dan 1 anders 0} \\
\mathsf{gt}(x,y) &= \text{als } x > y \text{ dan 1 anders 0} \\
\mathsf{le}(x,y) &= \text{als } x \leq y \text{ dan 1 anders 0} \\
\mathsf{ge}(x,y) &= \text{als } x \geq y \text{ dan 1 anders 0}
\end{aligned}$$

$$\begin{aligned}
\mathsf{eq}(x,y) &= \text{als } x = y \text{ dan 1 anders 0} \\
\mathsf{ne}(x,y) &= \text{als } x \neq y \text{ dan 1 anders 0} \\
\mathsf{max}(x,y) &= \text{het maximum van } x \text{ en } y \\
\mathsf{min}(x,y) &= \text{het minimum van } x \text{ en } y \\
\mathsf{quo}(x,y) &= \text{als } y \neq 0 \text{ dan } \lfloor x/y \rfloor \text{ anders 0} \\
\mathsf{rem}(x,y) &= \text{als } y \neq 0 \text{ dan } x \bmod y \text{ anders } x \\
\mathsf{divides}(x,y) &= \text{als } y \neq 0 \text{ en } y \,|\, x \text{ dan 1 anders 0} \\
\mathsf{even}(x) &= \text{als } x \text{ even is dan 1 anders 0} \\
\mathsf{prime}(x) &= \text{als } x \text{ priem is dan 1 anders 0} \\
\mathsf{pn}(x) &= \text{het } x\text{-de priemgetal} \\
& \text{(dus } \mathsf{pn}(0) = 2,\ \mathsf{pn}(1) = 3, \text{ etc.)}
\end{aligned}$$