Tilstandsmaskiner avslutning

Forrige gang: Vi så et eksplisitt eksempel på et språk over $\{0,1\}$ som <u>ikke</u> er regulært.

Vi kan også indirekte vise at det finnes språk som ikke er regulære.

<u>Lemma</u> La A være et alfabet (endelig). Da er A^* tellbar.

<u>Bevis</u>: Velg en bijektiv funksjon

$$A \xrightarrow{g} \{0,1,\dots,n-1\}$$

Altså slik at $|A| = n$.

Da er $f: A^* \to \mathbb{N}$ definert ved

$$f(sa) = f(s) \cdot n + g(a)$$

$$f(\Lambda) = 1$$

en injektiv funksjon.

Så med mindre $A = \emptyset$, og da $A^* = \{\Lambda\}$, er A^* en uendelig, tellbar mengde.

Da er $P(A^*)$ ikke tellbar. Hvor mange regulære språk finnes det over A?

Husk at et regulært uttrykk gir et regulært språk:

$$\text{Reg. uttrykk} \xmapsto{L} \text{Reg. språk}$$

Alle regulære språk beskrives av minst ett regulært uttrykk. Merk: To ulike uttrykk kan beskrive samme språk:

$$L(0^* \mid 000) = L(0^*)$$

Merk også at gitt et alfabet A er mengden av regulære uttrykk over A en delmengde av $(A \cup \{(,), \mid, *\})^*$.

La $S = A \cup \{(,), \mid, *\}$

A er endelig, så S er endelig. Da er S^* tellbar.

Altså er mengden av regulære uttrykk også tellbar. Det følger at mengden av regulære språk over A

også er tellbar. Men mengden av alle språk over A er ikke ~~A~~ tellbar. Dermed må det eksistere språk over A som ikke er regulære.

Å identifisere slike språk kan være vanskelig.

Forrige gang brukte vi Kleenes teorem og duehullprinsippet til å vise at $\{1^n 0^n \mid n \geq 1\}$ ikke er regulært.

Eventuelt: Snakke mer om dette beviset.

Vi finner det ~~[illegible]~~ språk $S \subseteq A^* = \{0,1\}^*$ slik at ingen datamaskin (selv med uendelig minne) kan gjenkjenne h/6

Kleenes Teorem

Vi har brukt Kleenes teorem en del allerede:

Et språk er regulært hvis og bare hvis det finnes en endelig tilstandsmaskin som gjenkjenner nøyaktig dette språket.

Implisitt: Vi har et fast, valgt alfabet i begge tilfeller.

Vi skal nå drøfte hvordan Kleene's teorem bevises uten å gå inn i alle detaljene.

Bevis-idé:

Det er to retninger å vise. La oss først drøfte implikasjonen

Regulært $\rightarrow$ Endelig tilstandsmaskin her er ideen å bruke strukturell induksjon. La oss for enkelhets skyld holde oss til alfabetet $A = \{0,1\}$

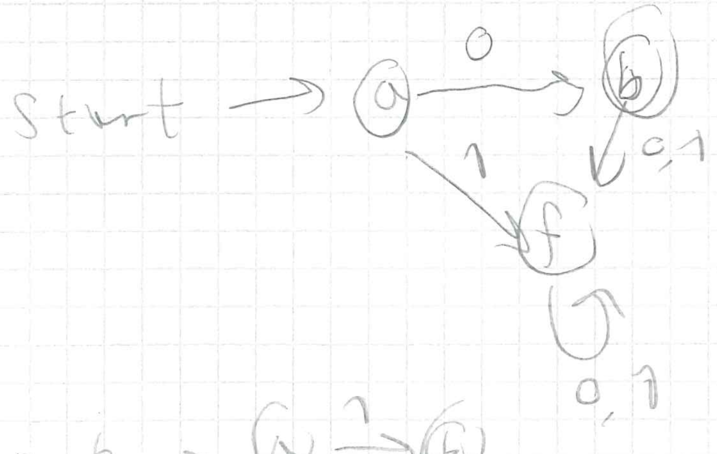
<u>Basis</u> $\emptyset$, $\{\lambda\}$, $\{0\}$ og $\{1\}$ er regulær

For hver av disse skal vi nå produsere en endelig tilstandsmaskin som gjenkjenner nøyaktig det lille språket:

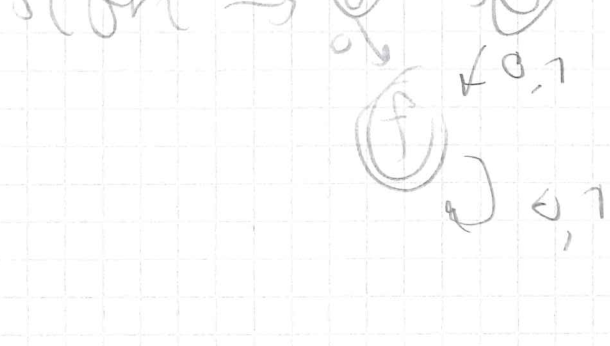
$\emptyset$:

$\{\lambda\}$:

$\{0\}$:



$\{1\}$:



Ind. Step

Vi må nå vise at hvis L, R er regulære språk slik at det finnes endelige tilstandsmaskiner som nøyaktig gjenkjenner dem, så finnes det også endelige tilstandsmaskiner som gjenkjenner

$$L^*$$

$$LR$$

$$L \cup R$$

De to første tilfellene er innviklede, så vi drøfter dem ikke her. Det tredje tilfellet kan vi derimot forsøke: h...

La T_1 og T_2 være tilstandsmaskiner som gjenkjenner nøyaktig L og R respektivt. Si at T_1 har tilstander $a_1, \ldots, a_n$ og T_2 har tilstander $b_1, \ldots, b_m$. Husk at $A = \{0, 1\}$.

~~La~~

La $f(x, i)$ være tilstanden vi blir sendt i når T_1 eller T_2 leser input i og er i tilstand x.

La M være tilstandsmaskinen med tilstander $\{a_1, \dots, a_n\} \times \{b_1, \dots, b_m\}$

En tilstand i M er altså et par

$$(x, y)$$

der x er tilstand i T_1 og y i T_2.

Når M er i tilstand (x, y) og leser input "0", flytter den seg til tilstand

$$(f(x, 0), f(y, 0))$$

Leser den derimot 1, flytter den seg til tilstand

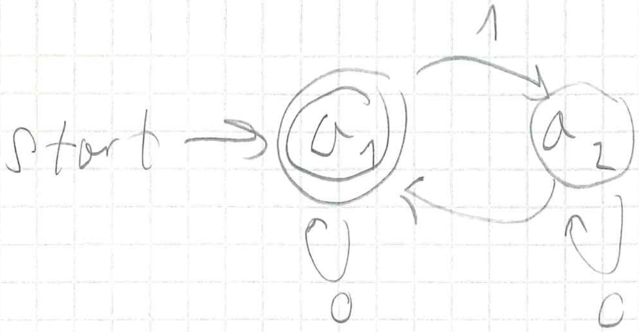
$$(f(x, 1), f(y, 1))$$

En tilstand (x, y) i M er aksepterende hvis x er aksepterende i M eller hvis y er aksepterende i Y, M starter i (start, start).

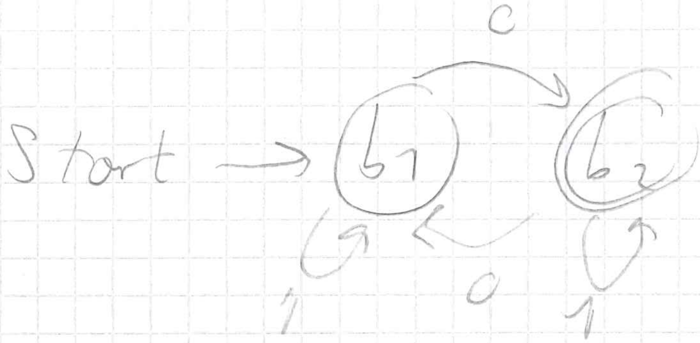
Da gjenkjenner M nøyaktig $L \cup R$.

Eks:

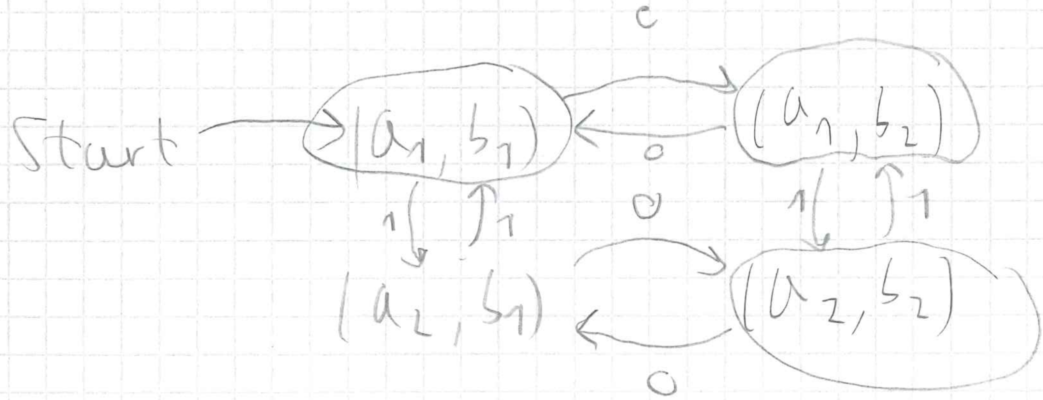
T_1:


start
a_1
a_2
1
0
0

T_2:


Start
b_1
b_2
0
0
1
1

M:


Start
(a_1, b_1)
(a_1, b_2)
(a_2, b_1)
(a_2, b_2)
0
0
0
0
1
1
1
1

Hadde vi også laget maskiner som hadde gjenkjent nøyaktig L^* og L^R hadde beviset vært komplett.

Vi nøyer oss med å si at dette kan gjøres, men er utenfor pensum.

Den "andre veien" må også bevises, altså:

Gitt en endelig tilstandsmaskin er det slik at språket av strenge den aksepterer er regulært.