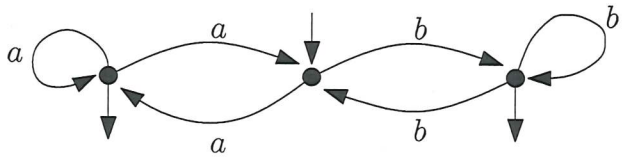
UPPSALA UNIVERSITET
Matematiska institutionen
Vera Koponen

TENTAMEN
Automatateori
2019-08-23

Skrivtid: 8 – 13. Tillåtna hjälpmedel: pennor, radergummi, linjal, papper (det sistnämnda tillhandahålles). *Betygsgränser:* För betyg 3/4/5 minst 18/25/32 poäng. Om man har fått minst 8 poäng, respektive minst 15 poäng, på duggan som gavs på hösten 2018 så får man uppgifterna 1 och 2, respektive 1 – 4, tillgodo (alltså full poäng utan att göra dem). *Om inget annat sägs så ska svaren/lösningarna motiveras på lämpligt sätt.*

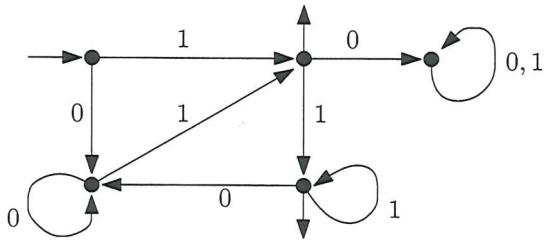
1. Konstruera en NFA (eller DFA) som accepterar språket som beskrivs av det reguljära uttrycket $ba^*(b(ab)^* \cup a^*)$. (3p)

2. Konstruera, med delmängdsalgoritmen, en DFA som accepterar samma språk som följande NFA: (3p)



3. Konstruera, med tillståndselimination, ett reguljärt uttryck för språket som accepteras av NFA:n i uppgift 2. (3p)

4. Konstruera, med särskiljandealgoritmen, en minimal DFA som accepterar samma språk som följande DFA. Om DFA:n redan är minimal så måste detta ändå motiveras med särskiljandealgoritmen. (3p)



5. Visa med hjälp av särskiljandesatsen eller pumpsatsen för reguljära språk och eventuellt slutenhetsegenskaper att följande språk *inte* är reguljärt: $L_1 = \{a^i b^j a^k : i \leq k\}$. (4p)

6. Nedan beskrivs en (restriktionsfri) grammatik G, där de stora bokstäverna är icke-terminerande symboler och de små är terminerande symboler. (4p)
(a) För var och en av strängarna 11111 och 1111 bestäm om den tillhör $L(G)$; om den tillhör $L(G)$ så visa en produktion av strängen, i annat fall förklara varför den inte kan produceras.
(b) Beskriv språket $L(G)$ med ord (och eventuellt matematisk notation) och motivera ditt svar.

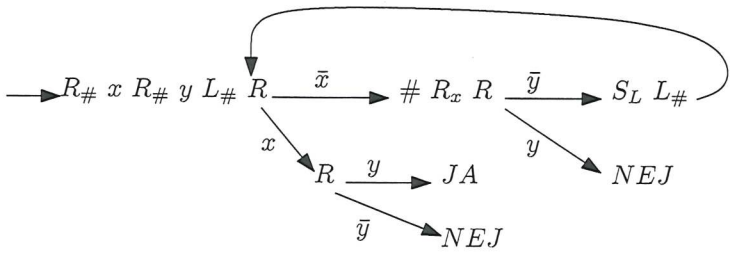
$$S \to Q1T \mid 1$$
$$T \to TT \mid P$$
$$1P \to P11$$
$$QP \to Q$$
$$Q \to \varepsilon$$

Fortsätter på nästa sida

7. Nedan beskrivs en TM vars inputalfabetet är $\{a, b\}$ och vars tapealfabet är $\{a, b, \#, x, y\}$. För varje tecken σ så betecknas "om inte σ avläses" med $\xrightarrow{\overline{\sigma}}$. Om vänstershiftaren S_L startas i konfigurationen $\#u\underset{\triangle}{\sigma}v\#$ (där σ är ett tecken) så stannar den i konfigurationen $\#uv\#$. Vi antar också att JA suddar tapen och sedan skriver ja, medan NEJ suddar tapen och sedan skriver nej. (4p)

(a) Kör maskinen på input $abb\#aaa$ och på input $ab\#aaa$ och notera speciellt om svaret ja eller nej ges.

(b) Vilken fråga besvarar maskinen? Med andra ord, vid start på $\#\underset{\triangle}{u}\#v$, när ges svaret ja och när ges svaret nej?



8. Finns det någon TM (dvs Turing-maskin) som (6p)

(a) givet godtyckliga TM:ar M_1 och M_2 avgör om M_1 och M_2 har samma språk (dvs accepterar exakt samma strängar)?

(b) givet godtyckliga TM:ar M_1 och M_2 avgör om M_1 och M_2 accepterar exakt samma strängar av längd högst 100?

9. För vart och ett av följande språk, bestäm om det är sammanhangsfritt eller inte. Kom ihåg att v^{rev} betecknar reverseringen av strängen v (alltså strängen baklänges). Svaren måste motiveras genom att en CFG eller PDA anges för språket om det är sammanhangsfritt, eller att pumpsatsen för sammanhangsfria språk används om detta inte är fallet. (6p)

$$L_2 = \{uuvv : u, v \in \{a, b\}^*\}$$
$$L_3 = \{uu^{rev}vv^{rev} : u, v \in \{a, b\}^*\}$$

10. Konstruera en (restriktionsfri) grammatik som producerar språket (4p)

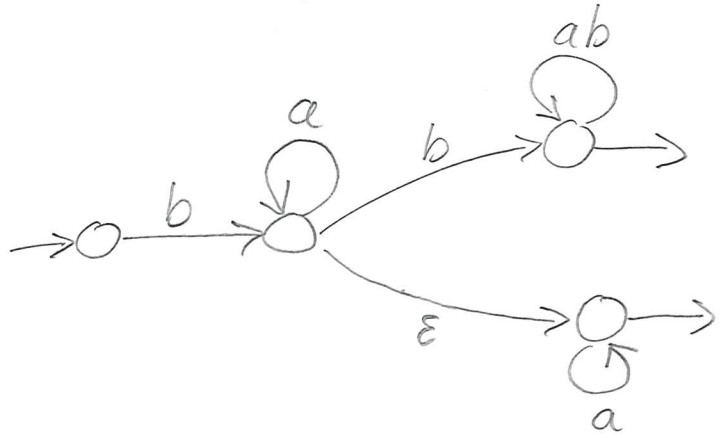
$$L_4 = \{w \in \{a, b\}^* : \; w \text{ innehåller exakt dubbelt så många } a \text{ som } b\}.$$

Lycka till!

Automatateori 2019-08-23

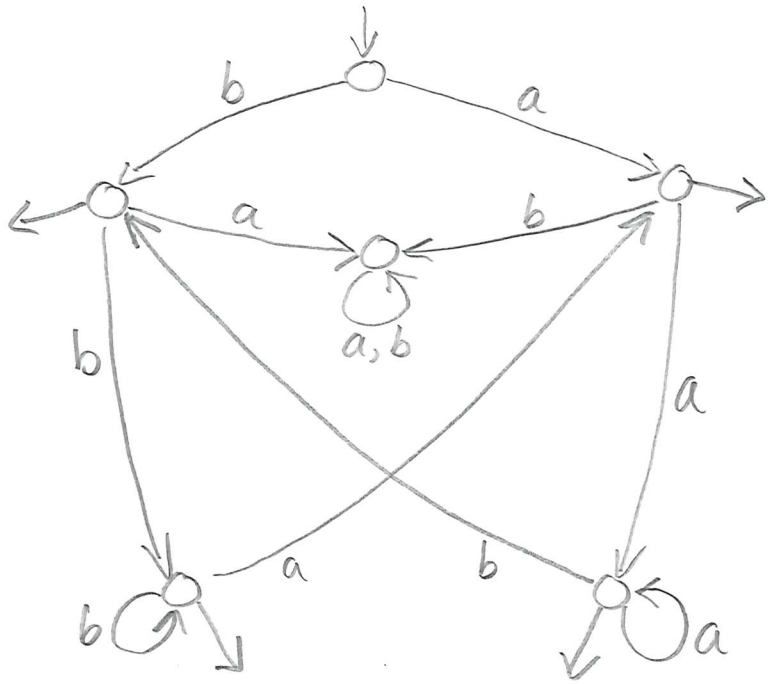
Svar / lösningsförslag

1.

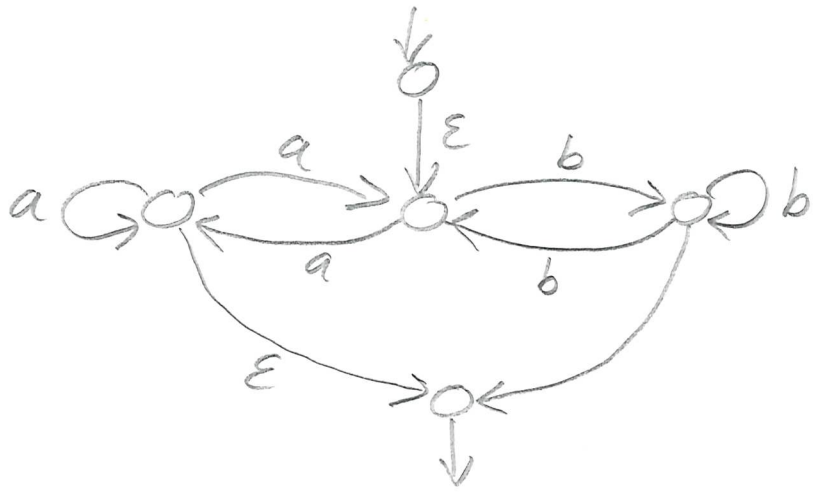


(Det finns andra NFA:er som accepterar samma språk.)

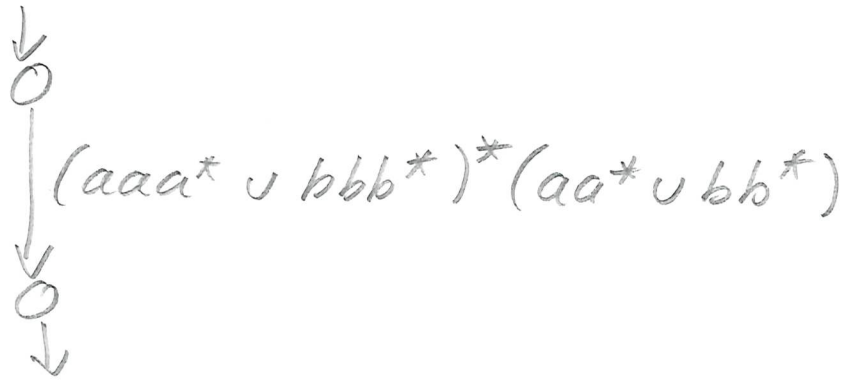
2.



3. Börja med att lägga till nytt start- och accepterande tillstånd.



Om man sedan eliminerar det vänstra tillståndet, sedan det högra och sist det mittersta, så får man



så $(aaa^* \cup bbb^*)^*(aa^* \cup bb^*)$ är ett reguljärt uttryck för NFA:ns språk.
(En fullständig lösning ska visa alla eliminationer, steg för steg.)

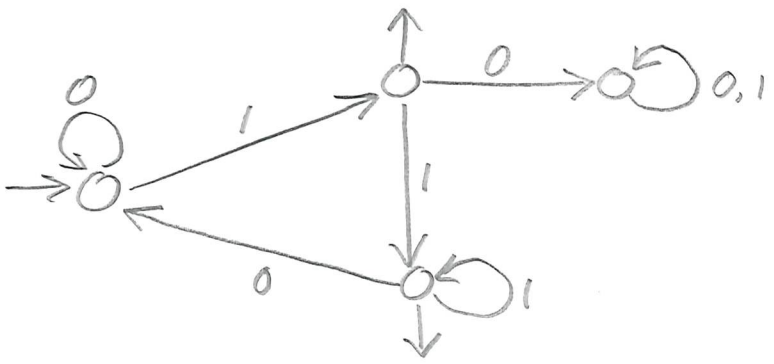
4. Om tillstånden numreras medurs med början i starttillstånder får man övergångstabellen

	1	2	3	4	5
0	5	3	3	5	5
1	2	4	3	4	2

Sedan används särskiljandealgoritmen.

Nivå	Uppdelningar	
0	{1, 2, 3, 4, 5}	
1	{2, 4} {1, 3, 5}	accepterande och icke-accept. särskiljs.
2	{2, 4} {1, 5} {3}	'1' driver DFA:n från 3 till 3 och från 1 till 2.
3	{2} {4} {1, 5} {3}	'0' driver DFA:n från 2 till 3 och från 4 till 5.
4	{2} {4} {1, 5} {3}	Går inte att sönderdela/ särskilja mer så vi är klara.

En minimal DFA fås genom att "slå ihop" tillstånden 1 och 5 (se nästa sida).



5. Med pumpsatsen:

1. L_1 är oändligt för tex. $a^1b^1a^1, a^2b^2a^2, a^3b^3a^3, \ldots \in L_1$.
2. Anta att L_1 är reguljärt.
3. Låt N vara given av pumpsatsen.
4. Låt $u = \varepsilon$, $w = a^N$, $v = ba^N$, så $|w| \geq N$ och $uwv = a^N b a^N \in L_1$.
5. Anta att $w = xyz$ och $y \neq \varepsilon$. Eftersom $y = a^m$ för något $m > 0$ så $uxy^2zv = a^{N+m} b a^N \notin L_1$.
6. Slutsatsen i punkt 5 motsäger pumpsatsen för reg. språk så L_1 kan inte vara reguljärt.

5. Med särskiljandesatsen:

Låt $A = \{a^i b : i = 1, 2, 3, 4, \ldots\}$ så A är oändlig. Om vi kan visa att L_1 särskiljer A så följer från särskiljandesatsen att L_1 inte är reguljär.

Låt $x, y \in A$ vara olika, så $x = a^n b$ och $y = a^m b$ för något n och m där $n \neq m$. Låt k vara det minsta talet av n och m och låt $z = a^k$. Då kommer exakt en av xz och yz att tillhöra L_1. Detta visar att L_1 särskiljer A (eftersom $x, y \in A$ var godtyckliga så att $x \neq y$).

6. (a) $S \Rightarrow Q1T \Rightarrow Q1TT$
$\Rightarrow Q1PT \Rightarrow QP11T \Rightarrow QP11P$
$\Rightarrow QP1P11 \Rightarrow QPP1111 \Rightarrow QP1111$
$\Rightarrow Q1111 \Rightarrow 1111$.

Så 1111 kan produceras (så den tillhör $L(G)$). Men 11111 kan inte produceras, så den tillhör inte $L(G)$. Detta pga av att

regeln $1P \to P11$ medför att inga strängar 1^n där $n>1$ och n är udda kan produceras.

(b) $L(G) = \{1^{2^n} : n = 0, 1, 2, 3, \ldots\}$.

Detta beror på att varje P som förflyttar sig till Q (som varje P måste om det ska kunna tas bort) fördubblar antalet ettor.

7. (a) Jag gör inte körningarna utan säger bara att om

TM:en får input abb#aaa så stannar den med svaret 'ja' och om

TM:en får input ab#aaa så stannar den med svaret 'nej'.

(b) TM:en besvarar, givet input u#v, frågan "Har u och v samma längd?".

godtycklig TM M_2 om $L(M_2) \in \Omega$, dvs. om $L(M_2) = \emptyset$.

(b) Svaret är nej. Motivation:

Antag att en TM M avgör, givet godtyckliga TM:ar M_1 och M_2, om M_1 och M_2 accepterar samma strängar av längd högst 100.
Låt M_1 vara en TM sådan att $L(M_1) = \emptyset$. Nu kan vi med M avgöra, för en godtycklig TM M_2, om M_2 accepterar någon sträng av längd högst 100. (Givet M_2 använd M för att avgöra om $L(M_1)$ och $L(M_2)$ innehåller samma strängar av längd ≤ 100.)
Låt $\Omega = \{L : L$ är TM-acceptabelt och innehåller ingen sträng av längd högst $100\}$

Med hjälp av Rices sats kan vi nu komma fram (jag hoppar över detaljerna) till att det inte finns någon TM som för godtycklig TM M_2 avgör om M_2

8. (a) Svaret är nej. Motivation:

Antag att det finns en TM M som avgör, för godtyckliga TM:ar M_1 och M_2 om $L(M_1) = L(M_2)$. Språket $\emptyset$ är TM-accepterbart så låt M_1 vara en TM sådan att $L(M_1) = \emptyset$.

Nu kan vi med M avgöra, för en godtycklig TM M_2 om $L(M_2) = L(M_1) = \emptyset$ (dvs. om $L(M_2) = \emptyset$).

Vi visar nu att detta motsäger Rices sats (så sådan M kan inte finnas).

Låt $\Omega = \{\emptyset\}$. Så

- Ω innehåller bara TM-accepterbara språk,
- Ω är inte tom och
- Ω innehåller inte alla TM-accepterbara språk.

Det följer nu från Rices sats att ingen TM kan avgöra för en

accepterar någon sträng av längd högst 100. Detta motsäger existensen av M i början av argumentet.

9. L_3 är sammanhangsfritt och här följer en CFG för L_3:

$$S \to UV$$
$$U \to aUa \mid bUb \mid \varepsilon$$
$$V \to aVa \mid bVb \mid \varepsilon$$

L_2 är inte sammanhangsfri.

Pumpsatsbevis: L_2 är oändlig eftersom alla $a^n a^n a^n a^n$, där $n = 0, 1, 2, \ldots$, tillhör L_2. Antag att L_2 är sammanhangsfritt. Låt k vara given av pumpsatsen för sammanhangsfria språk.
Välj $w = a^{k+1} b^{k+1} a^{k+1} a^{k+1} b^{k+1} a^{k+1}$. Om $u = a^{k+1} b^{k+1} a^{k+1}$ och $v = \varepsilon$ så $w = uuvv \in L_2$ och $|w| \geq k$. Antag att $w = uvxyz$ där $|vxy| \leq k$ och $vy \neq \varepsilon$. Då är vxy en delsträng av $a^{k+1} b^{k+1}$, $b^{k+1} a^{k+1}$ eller $a^{k+1} a^{k+1}$.

Därför (och eftersom $vy \neq \varepsilon$) gäller att

$uv^0xy^0z = uxz = a^{n_1}b^{m_1}a^{n_2}a^{n_3}b^{m_2}a^{n_4}$ där

$n_1, m_1, n_2, n_3, m_2, n_4 > 0$ och

$m_1 \neq m_2$ eller

en av n_1, n_2, n_3 och n_4 är mindre än $k+1$ och de tre andra är $k+1$.

På grund av detta kan vi <u>inte</u> ha

$$a^{n_1}b^{m_1}a^{n_2}a^{n_3}b^{m_2}a^{n_4} = u'u'.$$

Jag visar nu att det <u>inte</u> är möjligt att ha

$$a^{n_1}b^{m_1}a^{n_2}b^{n_3}b^{m_2}a^{n_4} = u'u'v'v'$$

där $u' \neq \varepsilon$ och $v' \neq \varepsilon$.

Eftersom $n_2 + n_3 > k+1$ så kan vi <u>inte</u> ha

$$b^{m_1}a^{n_2}a^{n_3}b^{m_2}a^{n_4} = v'v'.$$

Alltså måste u' sluta med 'b' (och börja med 'a'). Då måste vi ha

$$u'u' = a^{n_1}b^{m_1}a^{n_2}a^{n_3}b^{i}$$

där $0 < i \leq m_2$, $n_1 = n_2 + n_3$ och $m_1 = i$.
Men eftersom högst en av n_1, n_2 och n_3 är mindre än $k+1$ och alla är positiva så

så kan inte $n_1 = n_2 + n_3$ gälla.

Det följer att $uxz \notin L_2$, vilket motsäger pumpsatsen för sammanhangsfria språk.
Så L_2 är inte sammanhangsfri.

10. En grammatik som producerar L_4:

$S \to AASB \mid \varepsilon$ } skapar strängar på formen $A^{2n}B^n$.

$AB \to BA$
$BA \to AB$ } Ändrar ordningen på alla möjliga sätt.

$A \to a$
$B \to b$ } Ersätter icke-terminerande A, B med terminerande a, b.