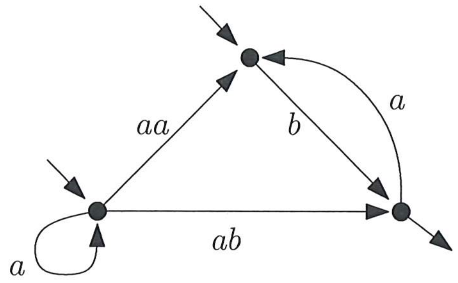
UPPSALA UNIVERSITET
Matematiska institutionen
Vera Koponen

DUGGA
Automatateori
2019-09-27

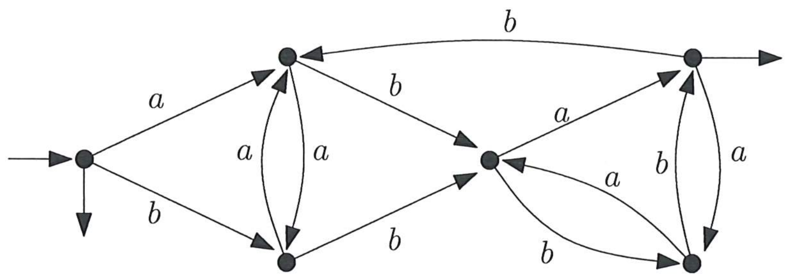
Skrivtid: 14 – 16. Tillåtna hjälpmedel: Bara pennor, radergummi, linjal och papper (det sistnämnda tillhandahålles). Varje uppgift 1–4 ger maximalt 5 poäng.

1. Konstruera, med delmängdsalgoritmen, en DFA som accepterar samma språk som följande NFA:



2. Konstruera, med tillståndselimination, ett reguljärt uttryck för språket som accepteras av NFA:n i uppgift 1.

3. 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.



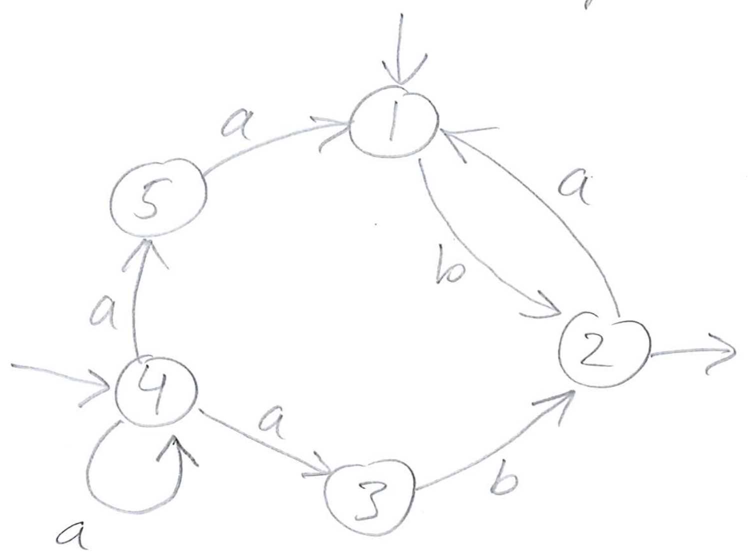
4. Kom ihåg att om $w = xy$ så kallas y för ett *suffix* till w. Bestäm för vart och ett av språken om det är reguljärt eller inte. Om det är reguljärt så ska det visas med hjälp av en NFA, DFA, reguljärt uttryck och/eller slutenhetsegenskaper; om det inte är reguljärt ska det visas med särskiljandesatsen eller pumpsatsen.

$$L_1 = \{w \in \{a,b\}^* : \text{ varje icketomt suffix till } w \text{ innehåller fler } b \text{ än } a\}$$
$$L_2 = \{w \in \{a,b\}^* : w \text{ har ingen förekomst av } aaa \text{ och ingen förekomst av } bbb\}$$

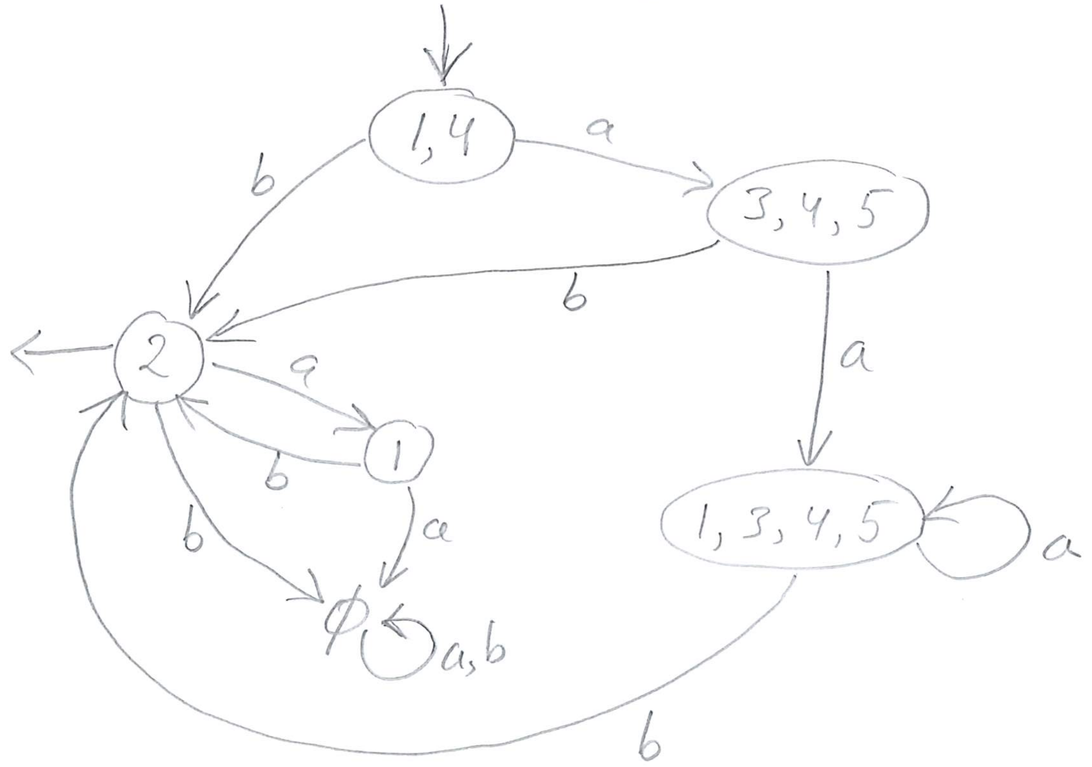
Lycka till!

Dugga 2019-09-27
Lösningsförslag

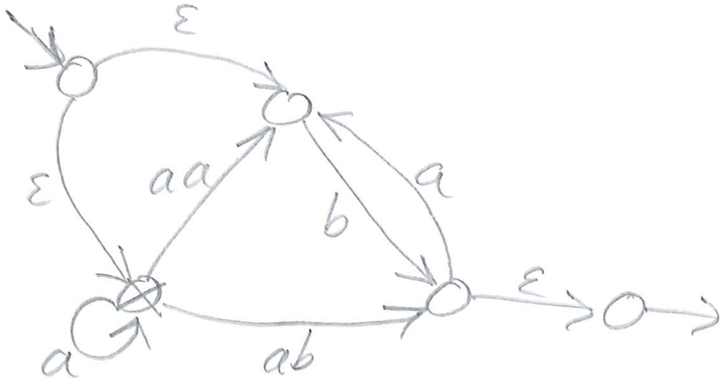
1. Först görs en icke-glupsk NFA som accepterar samma språk:



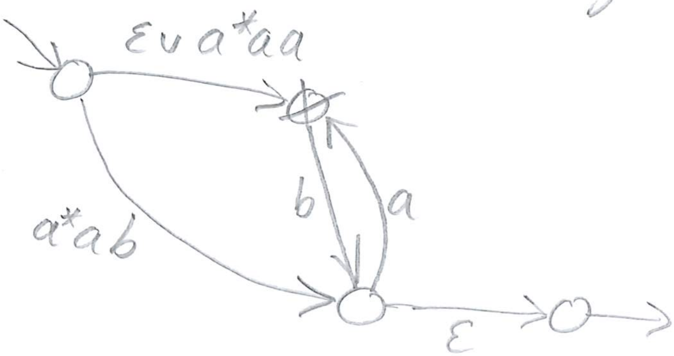
Sedan används delmängdsalgoritmen:

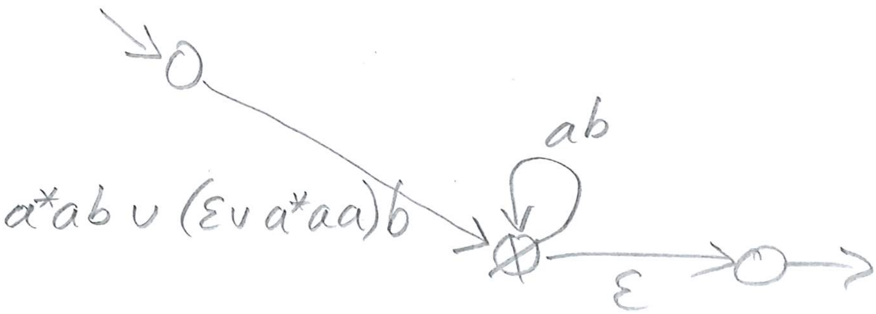


2. Nytt starttillstånd och nytt accepterande tillstånd läggs till:



Sedan elimineras alla de gamla tillstånden, ett för ett, och jag gör vissa förenklingar på en gång:

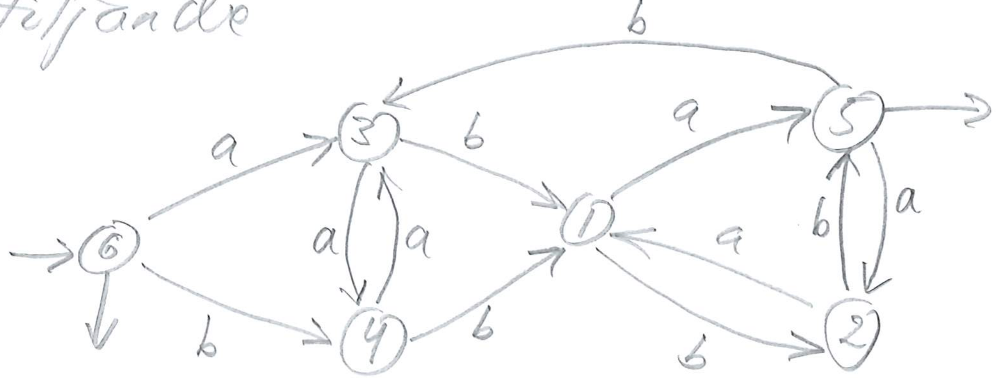




$(a^*ab \cup (\varepsilon \cup a^*aa)b)(ab)^*$

Detta är ett reguljärt uttryck för språket som accepteras av NFA:n.

3. Om jag numrerar tillstånden enligt följande



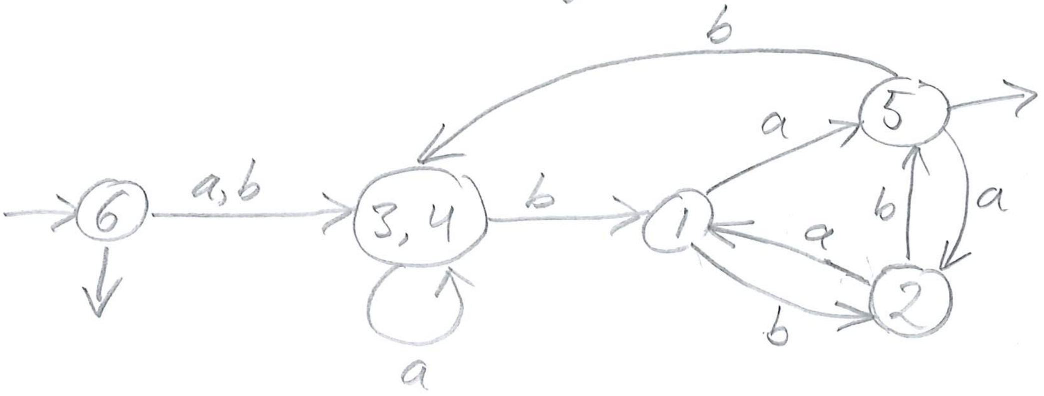
så kan övergångarna beskrivas av tabellen:

	1	2	3	4	5	6
a	5	1	4	3	2	3
b	2	5	1	1	3	4

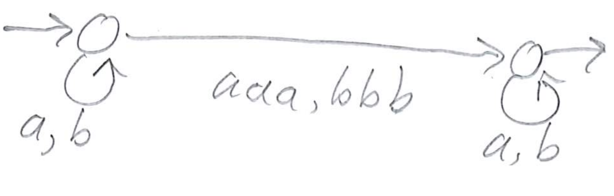
Vi använder nu särskiljandealgoritmen:

Nivåer	Sönderdelningar	
1	{1, 2, 3, 4} {5, 6}	De accepterande och icke-accepterande skiljs åt.
2	{1} {2} {3, 4} {5, 6}	'a' driver DFA:n från 1 till 5 och från 2 till 1. 'b' driver DFA:n från 2 till 5 och från 3 till 1.
3	{1} {2} {3, 4} {5} {6}	'a' driver DFA:n från 5 till 2 och från 6 till 3.
4	{1} {2} {3, 4} {5} {6}	Ingen mer sönderdelning kan göras.

En *minimal* DFA med samma språk som den ursprungliga DFA:n:



4. L_2 är reguljärt. För $\overline{L_2}$ är reguljärt eftersom $\overline{L_2}$ beskrivs av det reguljära uttrycket $(a \cup b)^*(aaa \cup bbb)(a \cup b)^*$.
(och accepteras av NFA:n



).

Eftersom $\overline{L_2}$ är reguljärt så är även $\overline{\overline{L_2}} = L_2$ reguljärt.

L_1 är *inte* reguljärt.
Bevis med pumpsatsen för reguljära språk:

1. L_1 är oändlig för $b^n \in L_1$ för alla $n \in \mathbb{N}$.

2. Antag att L_1 är reguljärt.

3. Låt N vara givet av pumpsatsen.

4. Låt $u = a^N$, $w = b^{N+1}$, $v = \varepsilon$, så $uwv = a^N b^{N+1} \in L_1$ och $|w| \geq N$.

5. Antag att $w = xyz$ och $y \neq \varepsilon$. Då finns $k \leq N$ så att $xz = b^k$ och det följer att $uxzv = a^N b^k \notin L_1$ eftersom $a^N b^k$ är ett suffix till $a^N b^k$. (Varje sträng är ett suffix till sig själv.)

6. Slutsatsen i punkt 5 motsäger pumpsatsen så L_1 är <u>inte</u> reguljärt.

Man kan också använda särskiljandesatsen och visa att någon <u>oändlig</u> mängd särskiljs av L_1. Man kan t.ex. välja den oändliga mängden $A = \{a^n : n \in \mathbb{N}\}$ och visa att den särskiljs av L_1.