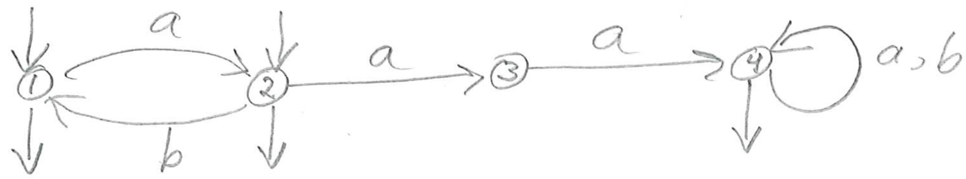
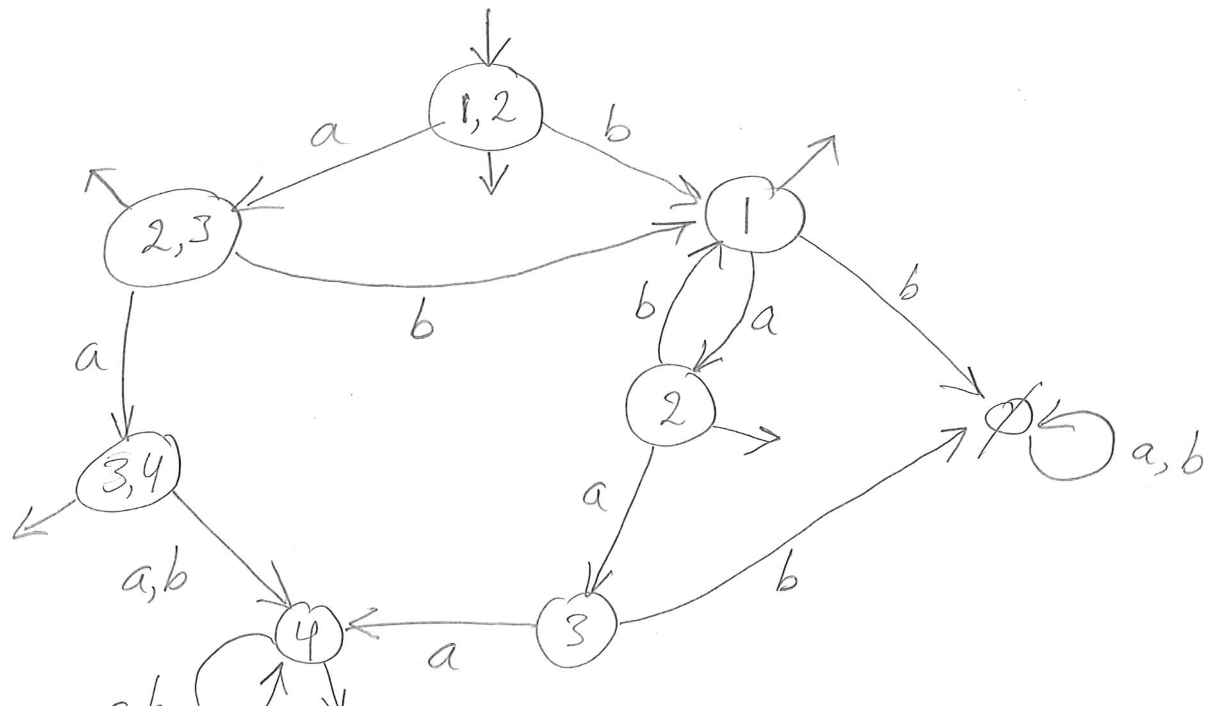
Lektion 2

Svar/lösningar

1. Först gör vi en icke-glupsk NFA som accepterar samma språk (och namnger tillstånden):

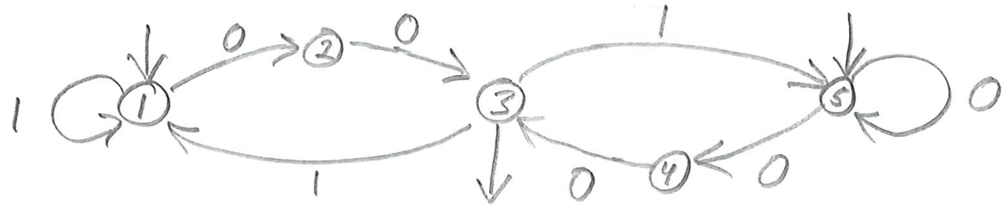


Sedan tillämpas delmängdskonstruktionen:

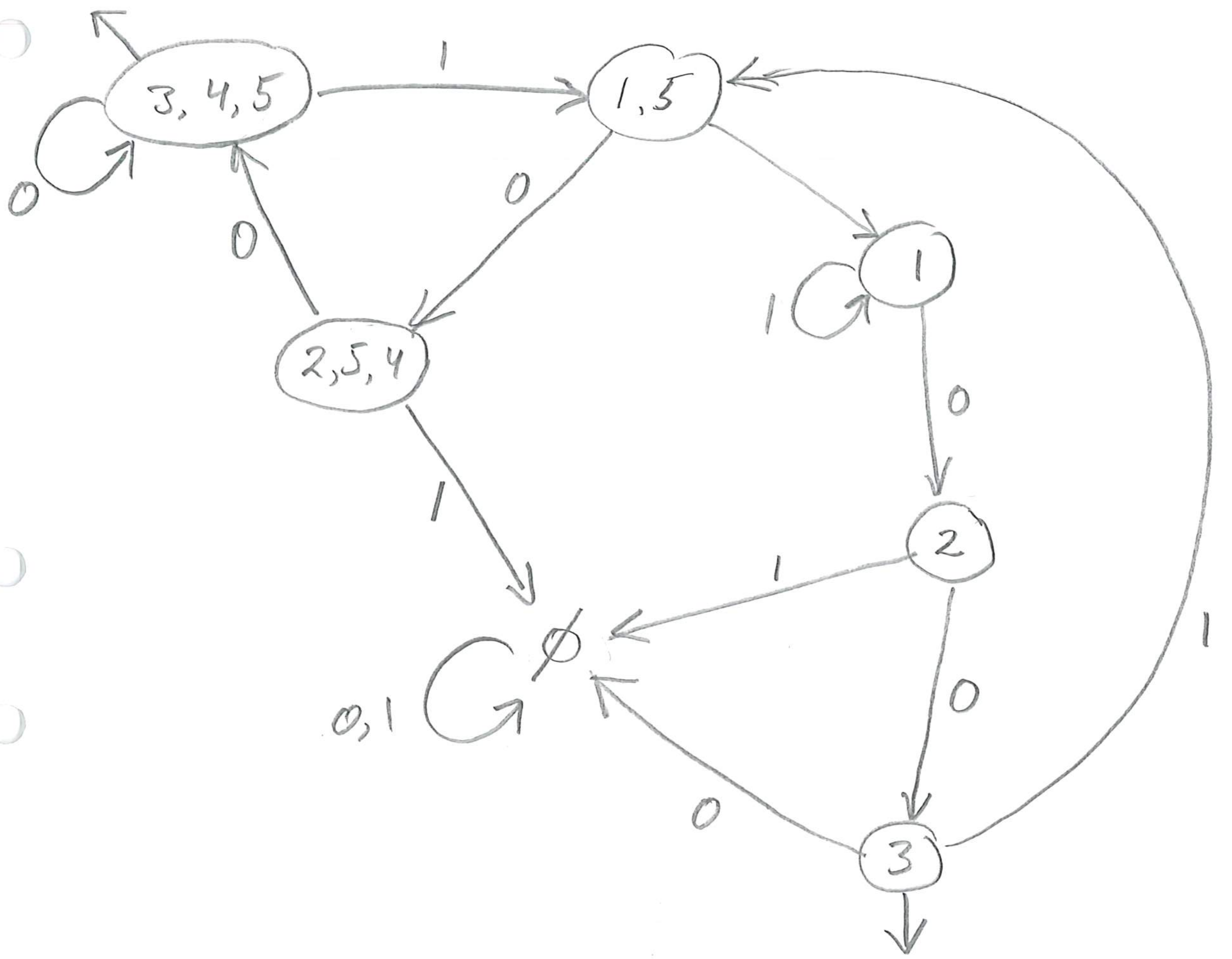


Nu har vi fått en DFA som accepterar samma språk som den ursprungliga NFA:n (och tillstånden i DFA:n kan få nya namn om vi vill).

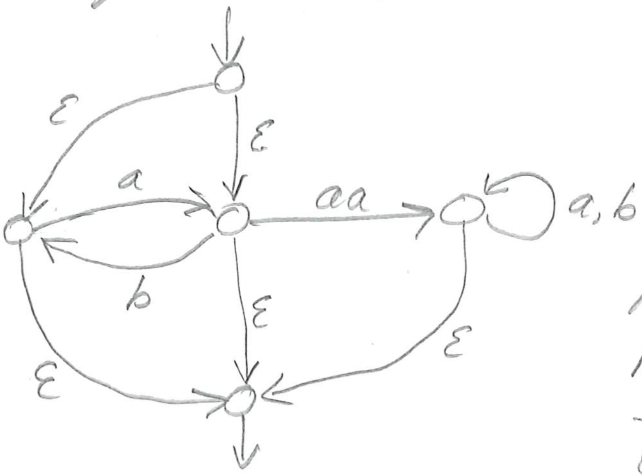
Först görs NFA:n icke-glupsk:



Tillstånden namnges och sedan tillämpas delmängdskonstruktionen:

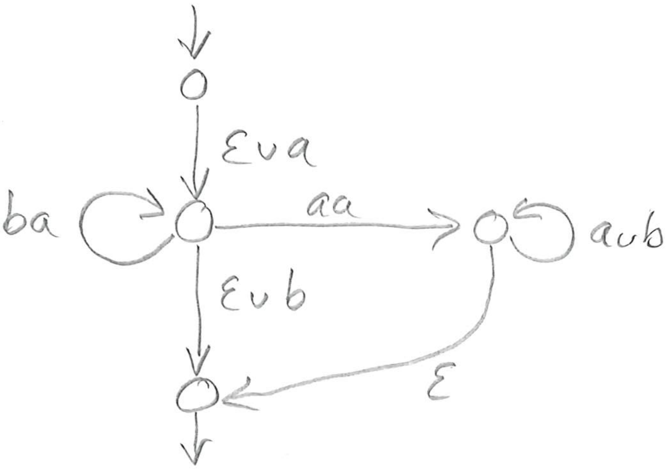


2. Först tilläggs ett nytt starttillstånd och ett nytt accepterande tillstånd:

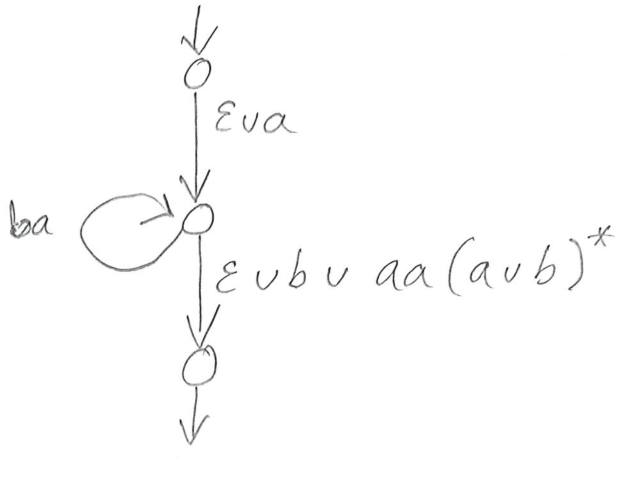


Notera att denna NFA accepterar samma strängar som den ursprungliga NFA:n.

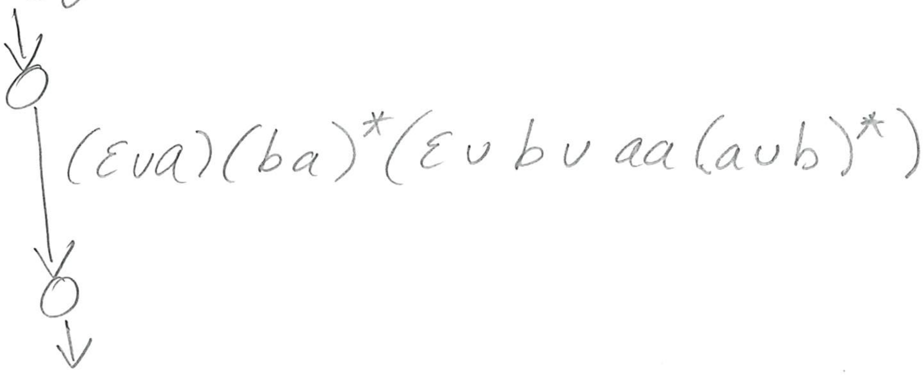
Sedan elimineras ett tillstånd (ej start- eller det accept. tillståndet):



Sedan elimineras ett till tillstånd:

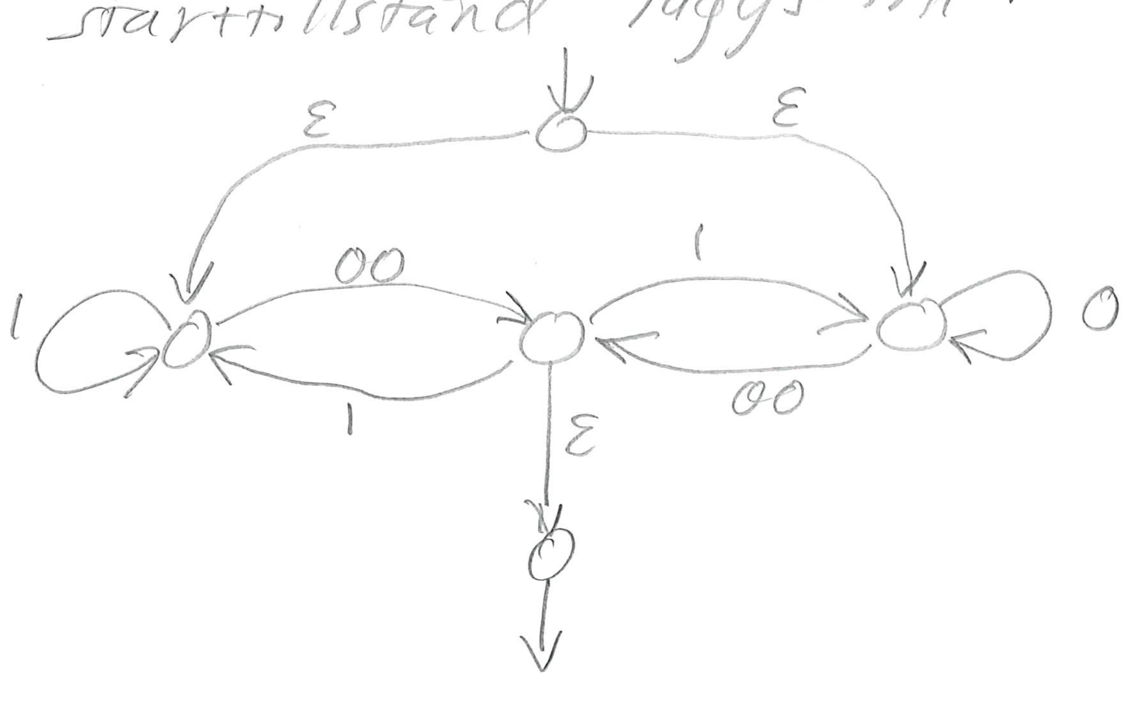


Och ytterliggare ett:

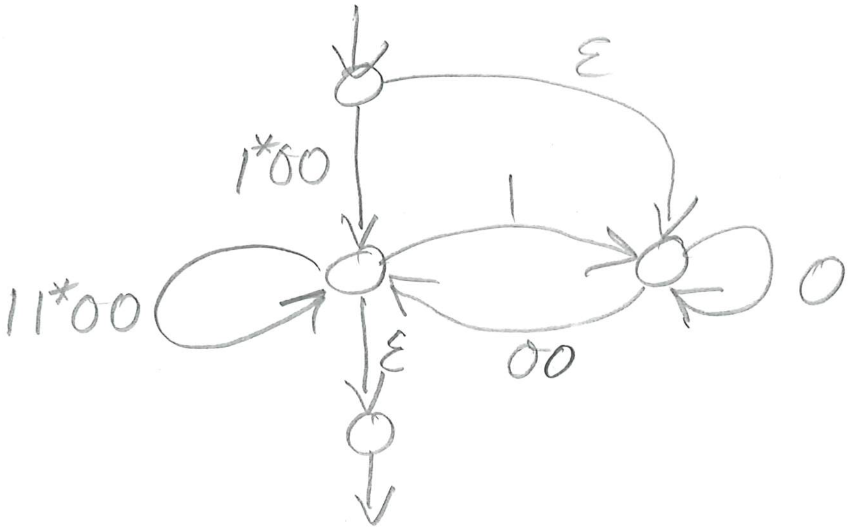


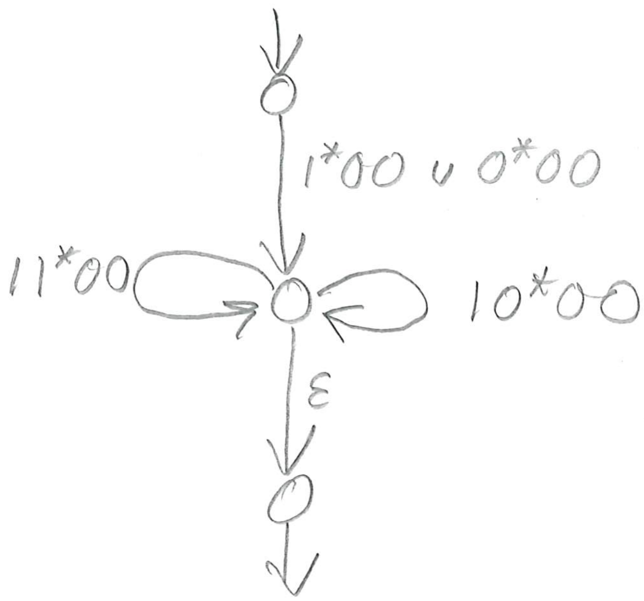
Nu har vi bara kvar det nya starttillståndet och det nya accepterande tillståndet (som aldrig ska elimineras). Uttrycket vid pilen från starttillståndet till det accepterande tillståndet är ett reguljärt uttryck för språket som den ursprungliga NFA:n accepterar. (Om man eliminerar tillstånden i en annan ordning så får man som regel ett annat uttryck.)

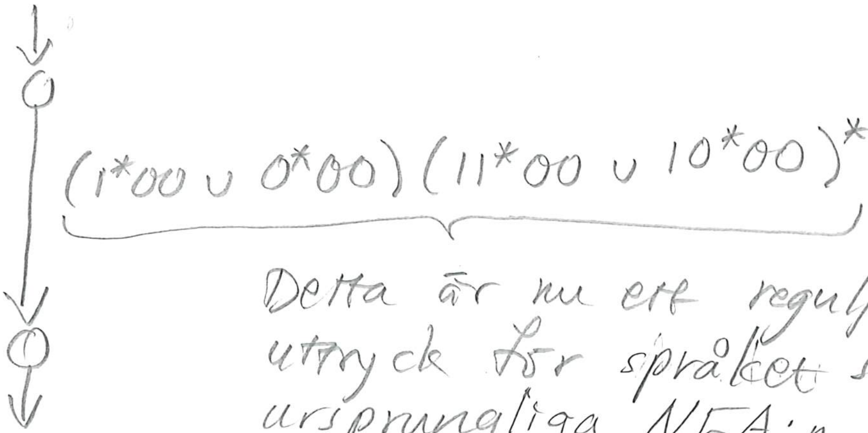
Nytt accepterande tillstånd och starttillstånd läggs till:



De tre ursprungliga tillstånden elimineras, ett för ett:







Detta är nu ett reguljärt uttryck för språket som den ursprungliga NFA:n accepterar.

3. I lösningarna till uppgift 6 till lektion 1 så ges DFA:er för alla tre språken. Man kan naturligtvis konstruera andra DFA:er för samma språk. Om man först konstruerar en NFA och sedan omvandlar den till en DFA så kanske man inte får samma DFA som om man konstruerar DFA:n direkt.

4. (a) Mängden av strängar över $\{a,b\}$ med högst tre stycken a:

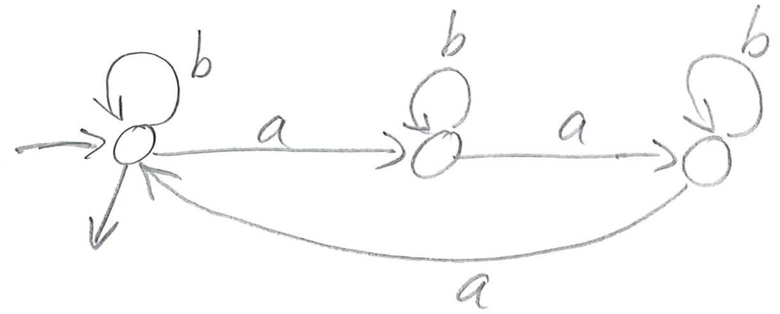
$$b^* \cup b^*ab^* \cup b^*ab^*ab^* \cup b^*ab^*ab^*ab^*$$

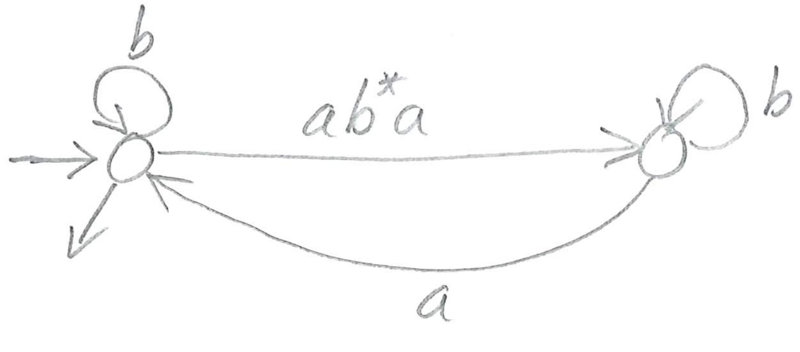
(c) Mängden av strängar över $\{a,b\}$ som har precis en förekomst av aaa:

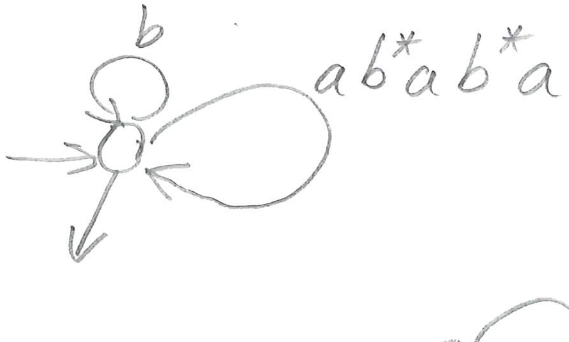
$$(b \cup ab \cup aab)^* aaa (b \cup ba \cup baa)^*$$

(b) Mängden av strängar över $\{a,b\}$ där antalet a är jämnt delbart med 3:

Jag utgår från följande DFA för språket och använder tillstånds-elimination. Eftersom DFA:n bara har ett start- och ett accepterande tillstånd som sammanfaller så eliminerar jag bara de övriga två tillstånden.





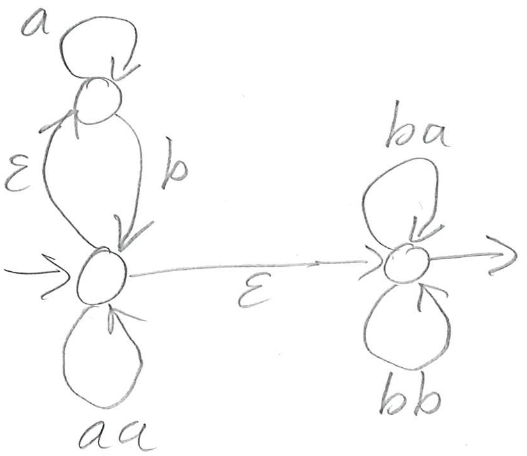


vilket förenklas till

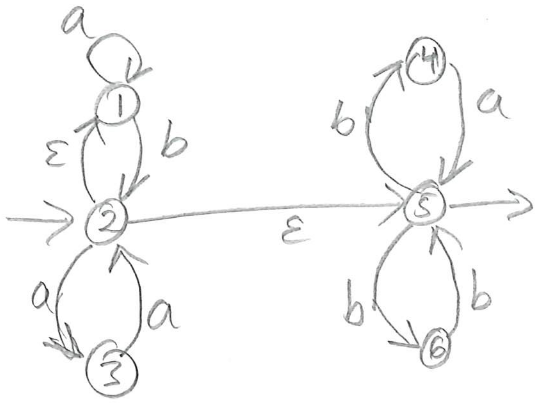
Ett reg. uttr. för språket är $(b \cup ab^*ab^*a)^*$.

5. Se svar/lösningar i kursboken.

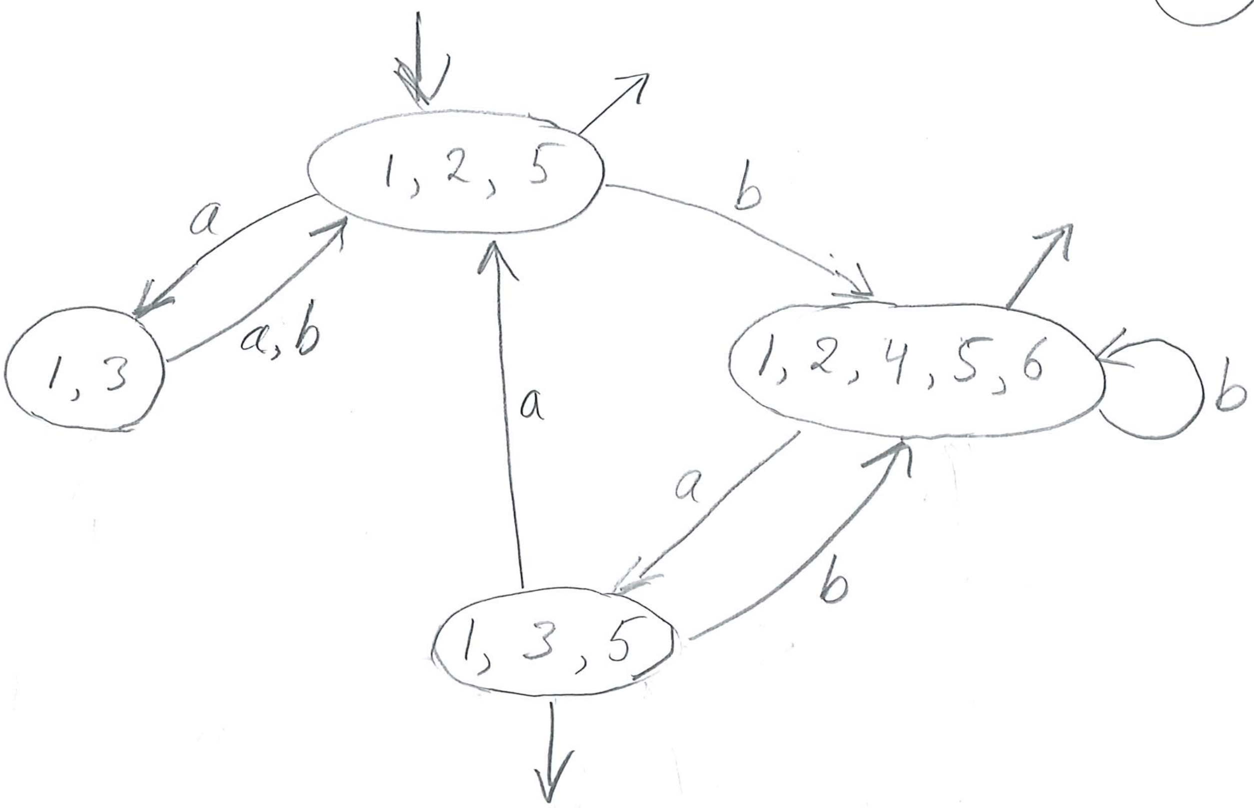
6. En NFA för $(a^*b \cup aa)^*(ba \cup bb)^*$:



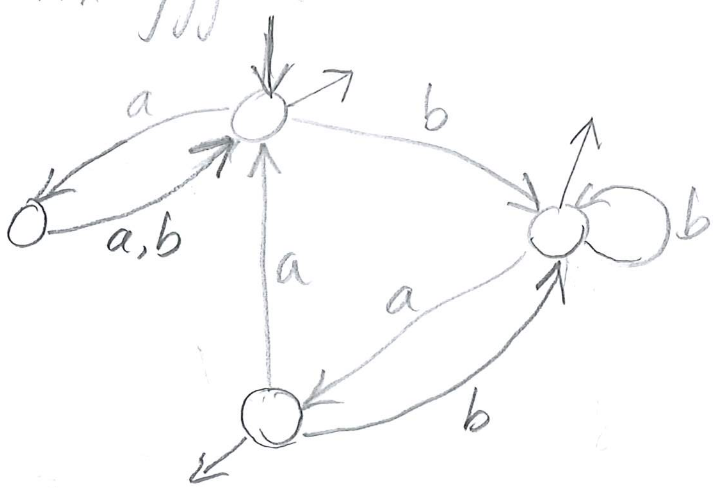
En icke-glupsk NFA som accepterar samma språk:

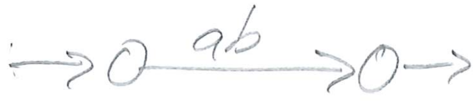


Nu använder vi delmängdsalgoritmen för att konstruera en DFA som accepterar samma språk.

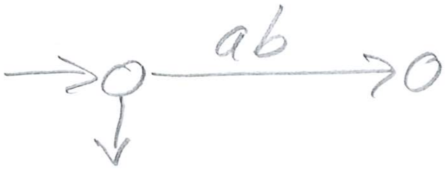


En "tillsnyggad" version:



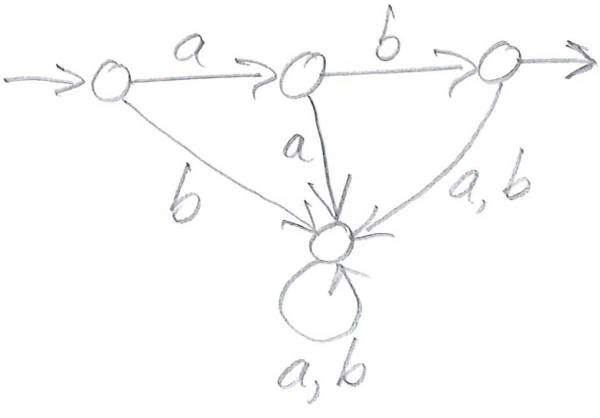
7. (a) NFA:n 

accepterar språket $L = \{ab\}$.

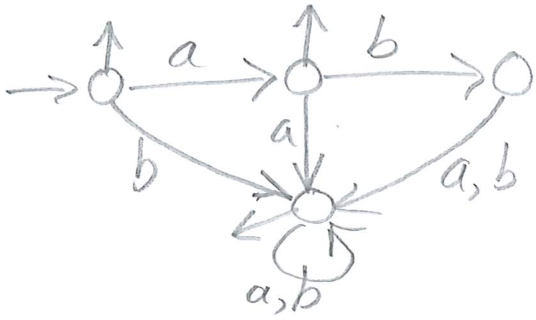
(b) NFA:n 

accepterar språket $\{\varepsilon\}$ vilket <u>inte</u> är samma som $\bar{L}$.
För $\bar{L}$ innehåller alla strängar över $\{a, b\}$ utom ab, men $\{\varepsilon\}$ innehåller bara strängen ε.

(c) En DFA som accepterar $L = \{ab\}$:

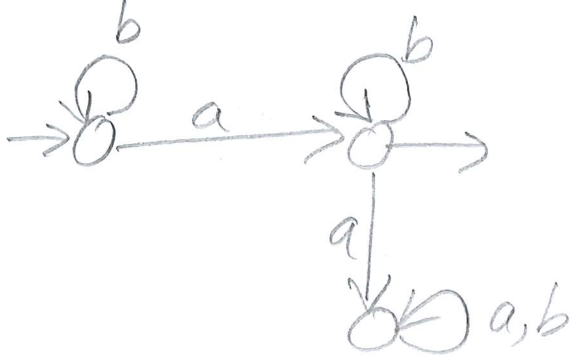


(d) En DFA som accepterar $\bar{L}$:



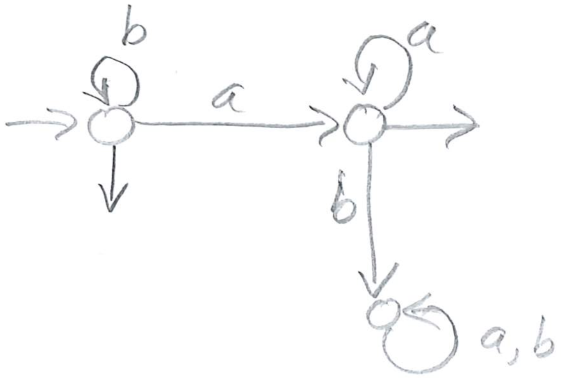
(Den enda strängen som inte accepteras är ab.)

8. (a) Låt M_1 vara



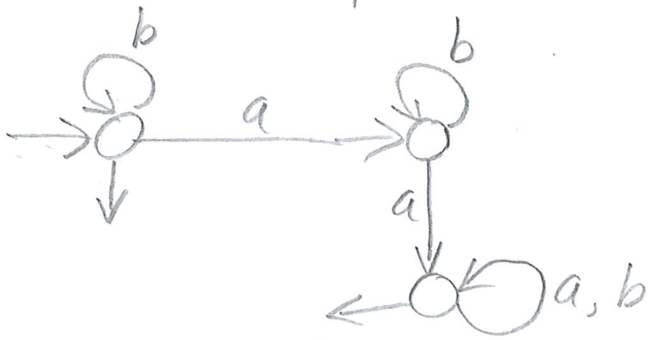
Då gäller $L(M_1) = L_1$.

(b) Låt M_2 vara

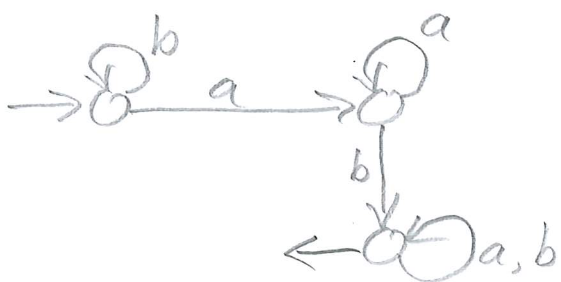


Då gäller $L(M_2) = L_2$.

(c) Om $\overline{M_1}$ är

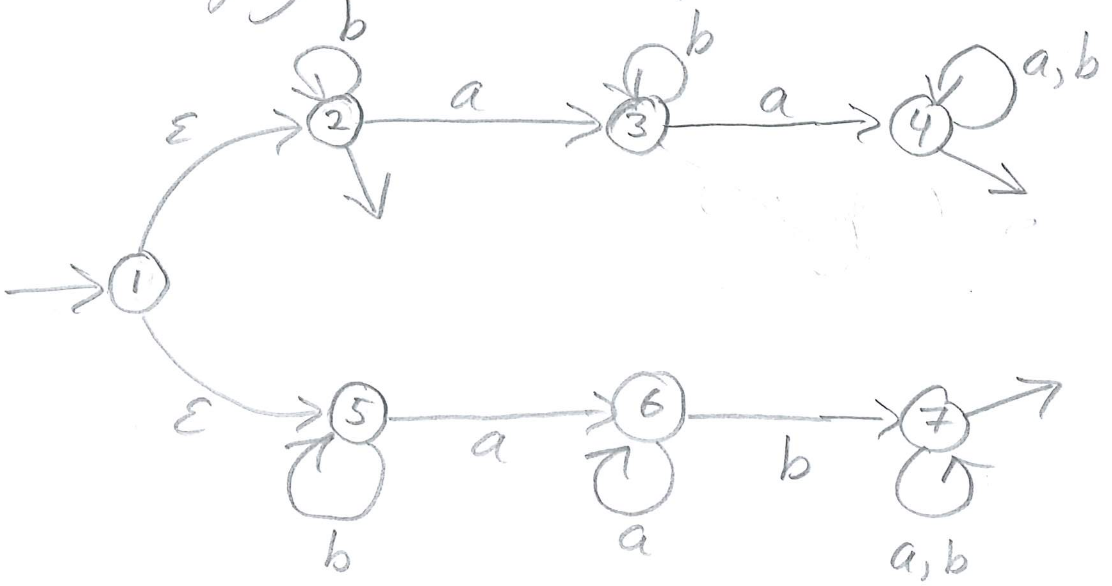


och $\overline{M_2}$ är

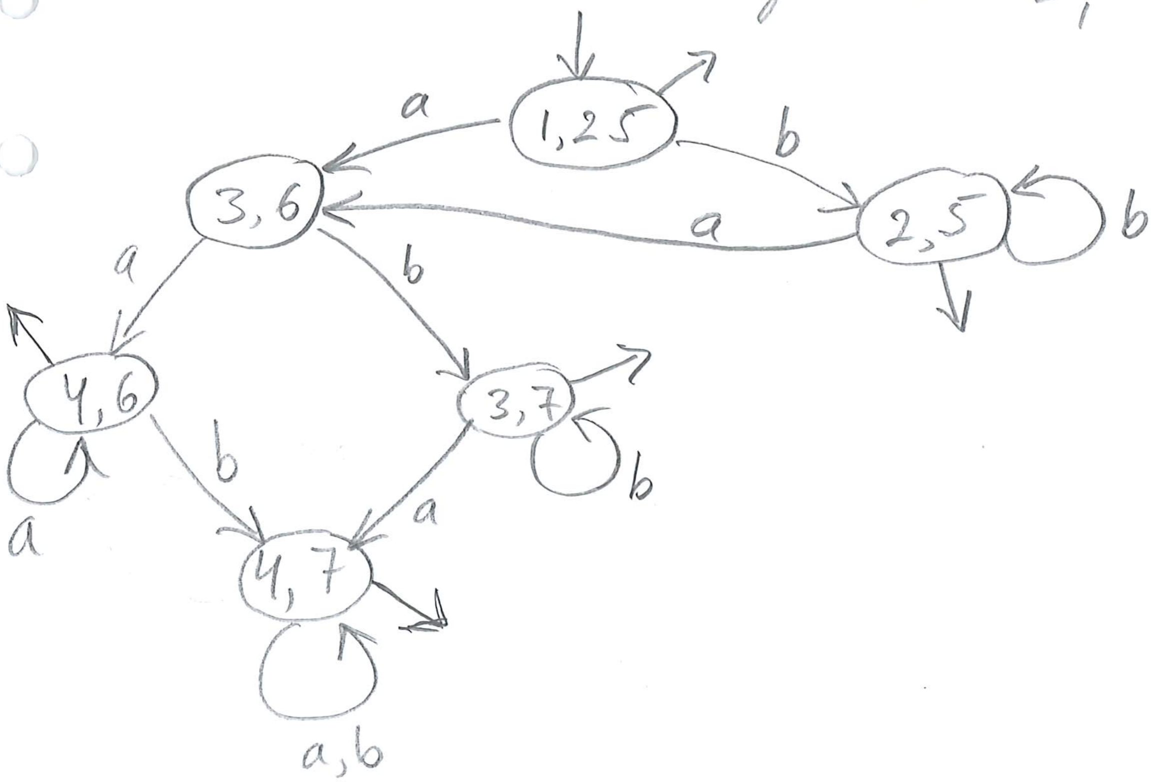


så $L(\overline{M}_1) = \overline{L}_1$ och $L(\overline{M}_2) = \overline{L}_2$.

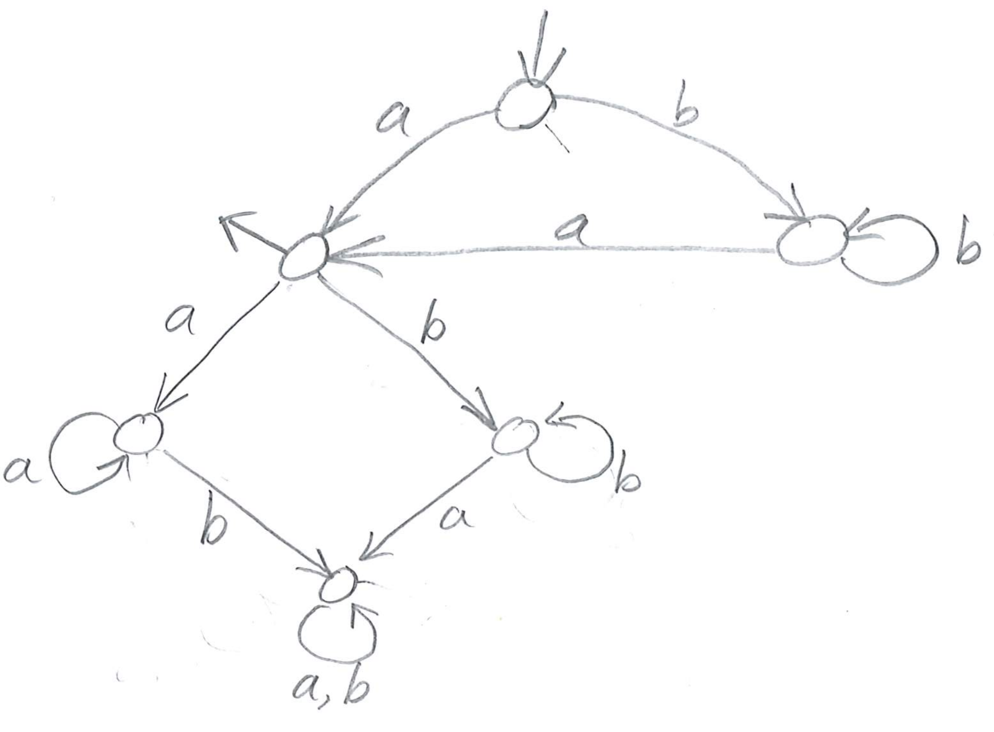
(d) En NFA som accepterar $\overline{L}_1 \cup \overline{L}_2$ där jag har namngivit tillstånden:



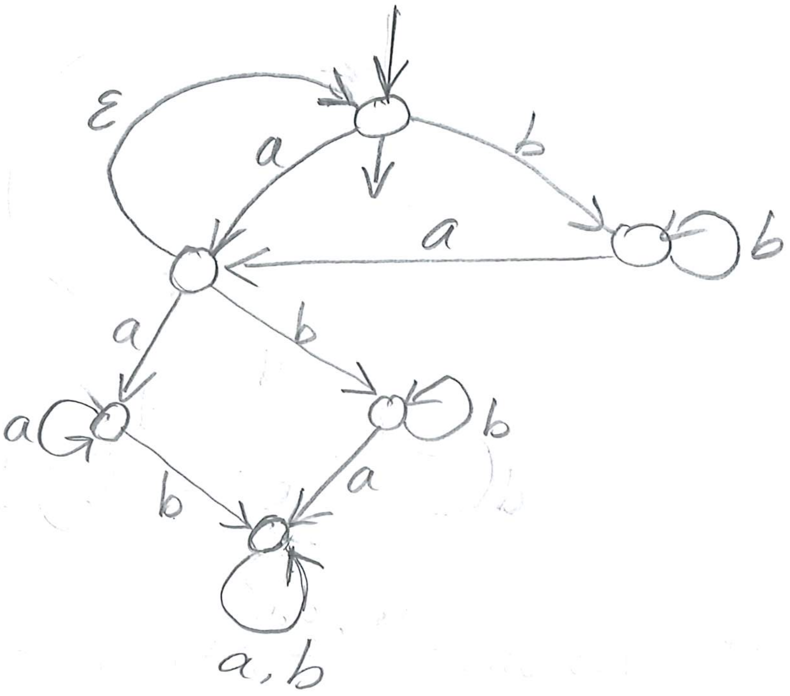
Med delmängdskonstruktionen får vi en DFA som accepterar $\overline{L}_1 \cup \overline{L}_2$:



(e) Eftersom $L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$ så får vi en DFA som accepterar $L_1 \cap L_2$ genom att ta DFA:n från del (d) och ändra så de accepterande tillstånden blir icke-accepterande och de icke-accepterande blir accepterande:

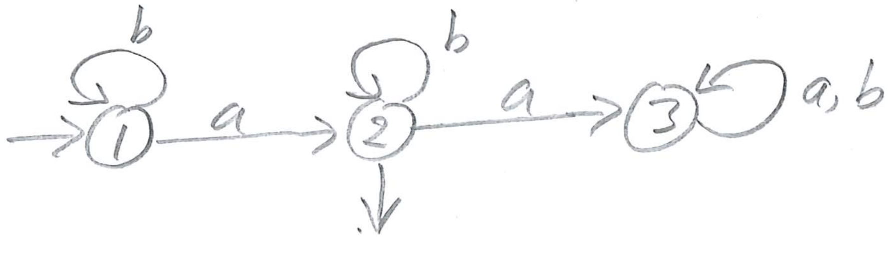


(f) Vi får en NFA som accepterar $(b^*ab^* \cap b^*a^*)^*$, dvs. $(L_1 \cap L_2)^*$ genom att "återkoppla" alla accepterande tillstånd till alla starttillstånd och göra så att alla starttillstånd blir accepterande.

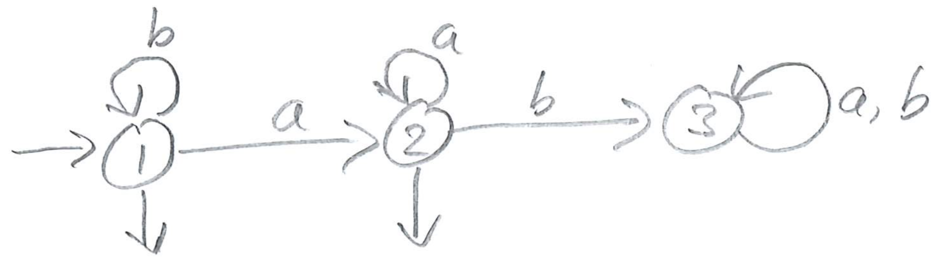


9. En DFA för $L_1 \cap L_2$ med en annan metod. Vi använder DFA:erna

M_1 :



M_2 :

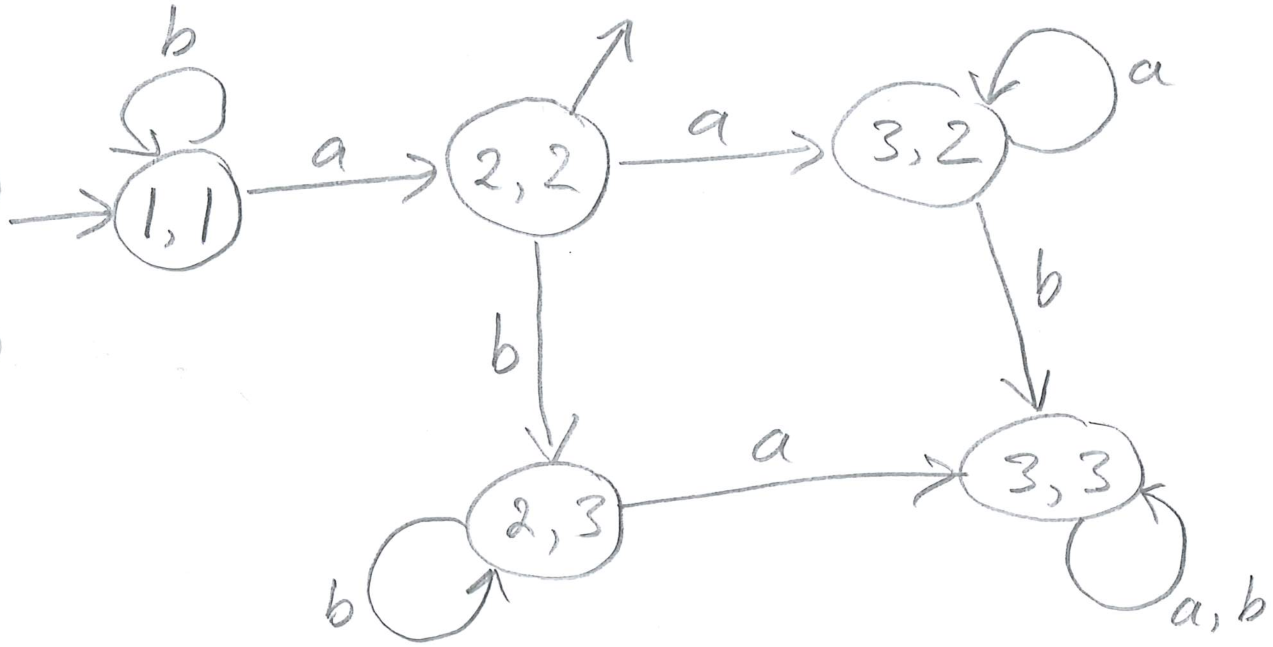


där $L(M_1) = L_1$ och $L(M_2) = L_2$.

Tillstånden för vår nya DFA M blir par (n, m) där n och m är tillstånd i M_1 och M_2, respektive. Vi kommer att ha en tillståndsövergång $(n, m) \xrightarrow{\sigma} (i, j)$ om det finns övergångar

$n \xrightarrow{\sigma} i$ i M_1 och

$m \xrightarrow{\sigma} j$ i M_2.



I den nya DFA:n är (n, m) ett starttillstånd om n och m är starttillstånd i M_1 och M_2, respektive och (n, m) är accepterande och n och m är accepterande i M_1 och M_2, respektive.