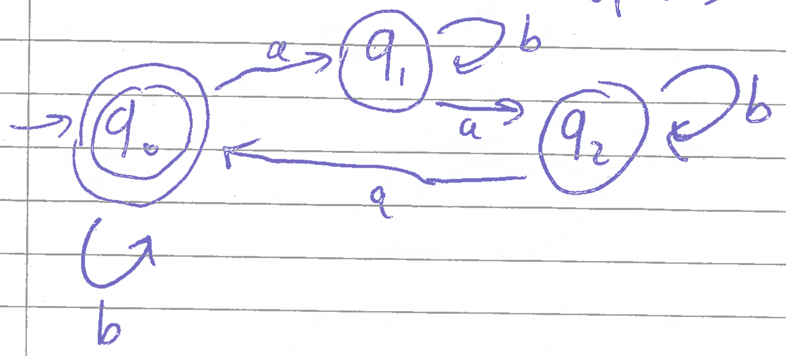
1/2 Stijn v/d Beemt s4331354 1C 8

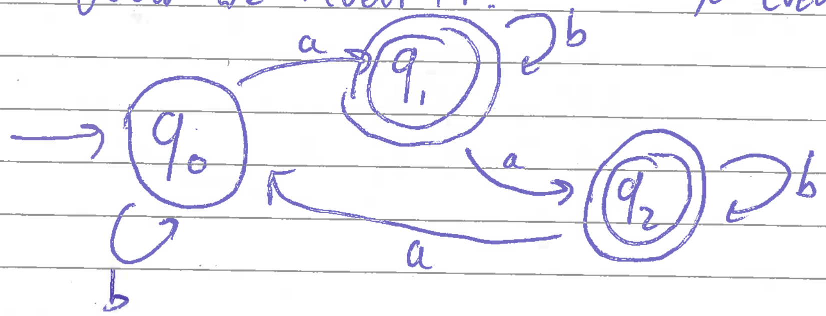
TuA ex2

First we'll define an automaton that produces words whose number of a's is only divisible by 3



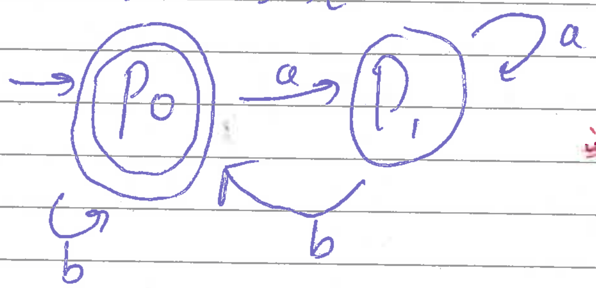
Now we invert it: We get everything that is not div by 3

1

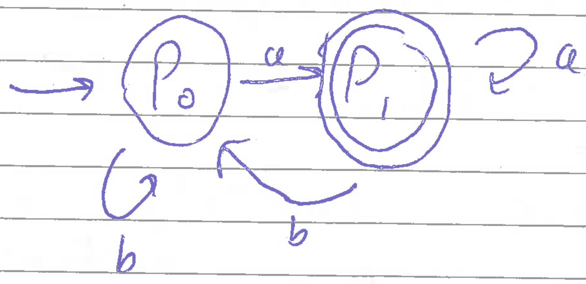


good

Now we create an automaton that always end with a b or is λ:



Now we invert it so it only accepts a:

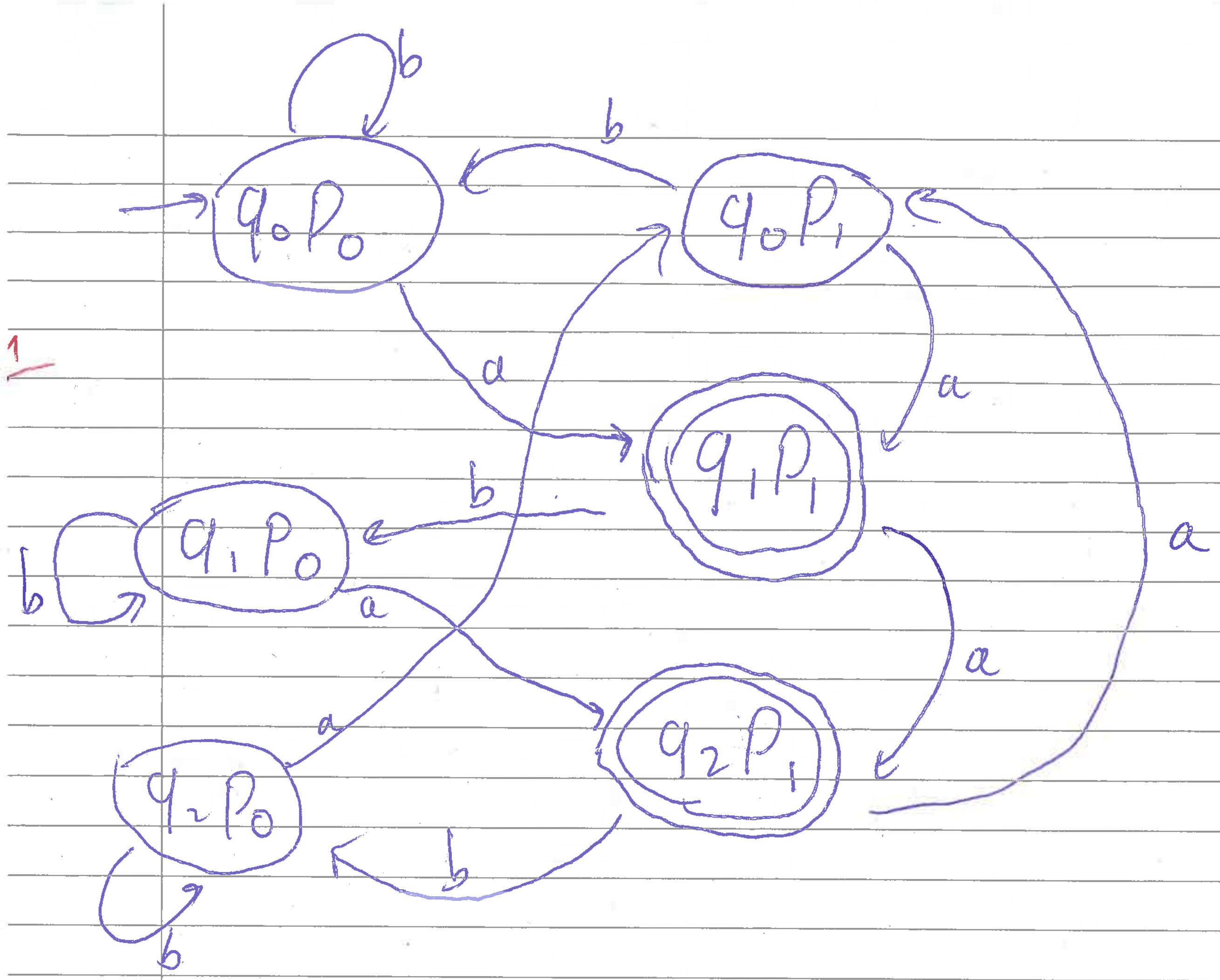


good

1

Now we create cartesian product automaton:

202



1

b abaa will not we accepted since it ends up in q_0P_1 which is not an accepting state.
0.5 $\longrightarrow \delta^*(q_0, abaa)$

0.5 ba will be accepted: it ends in q_1P_1 which is an accepting state
$\longrightarrow \delta^*(q_0, ba)$

c L_1 with $F_1 = \emptyset$ only accepts O
1 so $L_1 = \{O\}$ ok correct notation $L_1 = \emptyset$

0 L_2 with $F_2 = \{q_0\}$ is $1 + (a^*b^*)^* = (a^*b^*)^*$
no, since abbaabbbb $\in L_2$ but not in your regular expression!

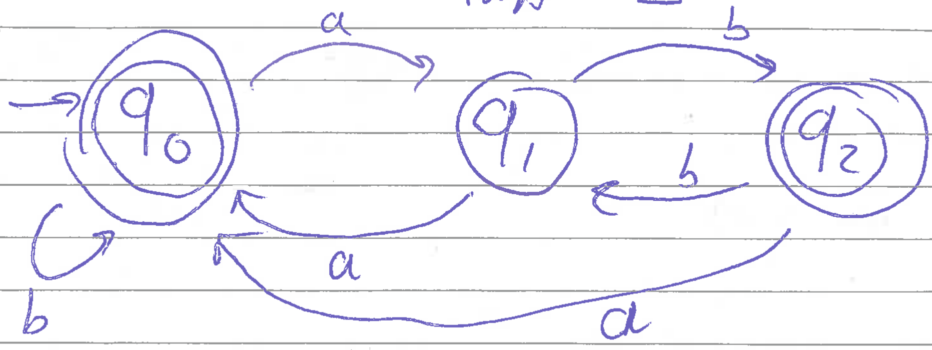
0 L_3 with $F_3 = \{q_1; q_2\}$ is $(a^*b^*)^*$
no counterexample: abbaa

$L_2 = \{w \in A^* \mid |w|_a, |w|_b$ are both even$\}$

$L_3 = \{w \in A^* \mid |w|$ is odd$\}$

2/2 Thijs vd Beemt 54331354 1C

TnA ex2

a) 

1

b $\cancel{((b^*(aa)^*)^*(abba)^*ab(bb)^*(baab)^*ba))^*}$

$\cancel{((b^*(aa)^*)^*(abba)^*ab(bb)^*(baab)^*ba)^*)^*}$

$\cancel{((b^*(aa)^*)^*(abba)^*ab(bb)^*(baab)^*ba))^*)^*(abba)^*ab}$

2

$$((b^*(aa)^*)^*(abba)^*ab(bb)^*((baab)^*ba + a))^*)^* +$$
$$((b^*(aa)^*)^*(abba)^*ab(bb)^*((baab)^*ba + a))^*)^*(abba)^*ab$$

you have indeed some redundancies.

I think this is a little too much...

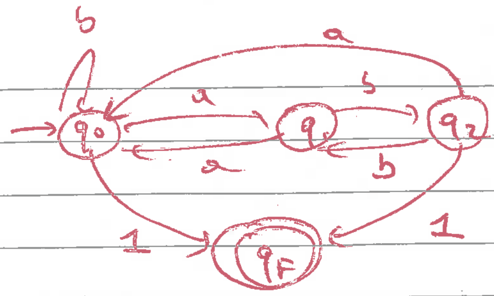
Would love some proper explanation on how to solve these

But this solution works, since I couldn't find any counterexample!

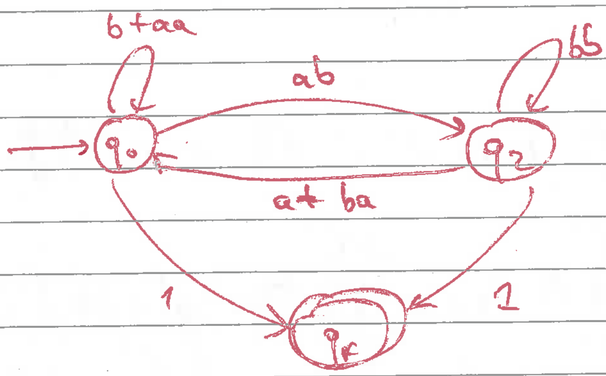
→ see verso

4 steps :

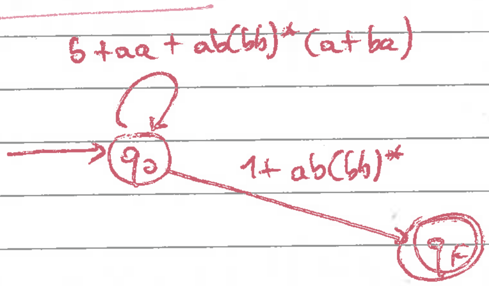
First, turn M into an automaton with one final state (special case - slide 22/23)



Then, collapse q_1 (slide 21/23 22/23)



Then collapse p_2



which gives $e = (b + aa + ab(bb)^*(a + ba))^* \; (1 + ab(bb)^*)$.

□