Assignment 2 Sample Solution

1. See attached diagram.

2.

   (a) All words that contain the substring $aaba$. Regular expression: $(\mathbf{a}+\mathbf{b})^*\mathbf{aaba}(\mathbf{a}+\mathbf{b})^*$.

   (b) All words that begin with a and end with b, and the empty word. Regular expression: $\mathbf{\Lambda}+\mathbf{a}(\mathbf{a}+\mathbf{b})^*\mathbf{b}$.

3.

   (a) If there are edges out of the final state by which a string can reach the same or another plus state, the new machine will accept not only the old language L and ab but also the concatenation of ab and all words that can be formed by concatenating the labels on the path from the final state to itself or another final state. See the first diagram on page 2 of the assignment for an example.

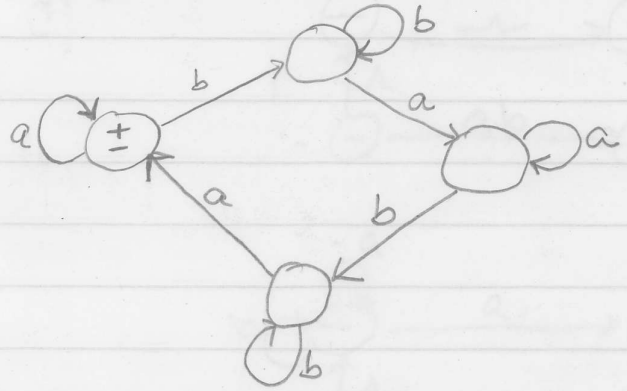
   (b) If there is a circuit that goes back to the start state, we face a problem similar to the one above: the machine will accept L and ab and also the product of any strings that can finish their processing in the start state concatenated with the string ab. See the second diagram on page 2 of the assignment for an example.

   (c) Add a new start state and a new final state to the machine and connect them with a transtion labeled ab. Connect the minus state to another minus state by a λ-edge. See the attached diagrams showing how to fix the above 2 examples.

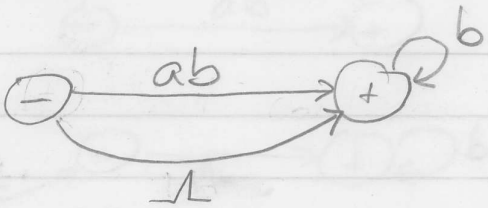
4. All words made up of 0 or more factors, where each factor is either the string $abbab$ or a substring beginning and ending with b with only a's in between. Regular expression $(\mathbf{ba}^*\mathbf{b}+\mathbf{abbab})^*$.

Assignment 2 Sample Solution

1.

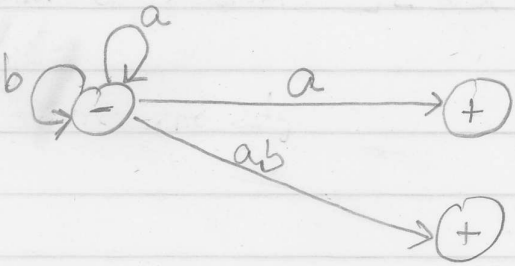


3.a)



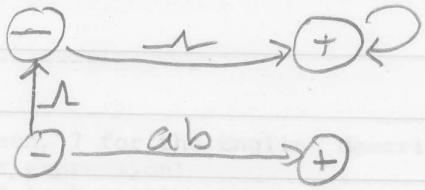
The TG before the ab edge was added accepted the language of all words made up of only b's (0 or more), which is b^*. The new language has more than ab added to it. It also has ab followed by any number of b's.

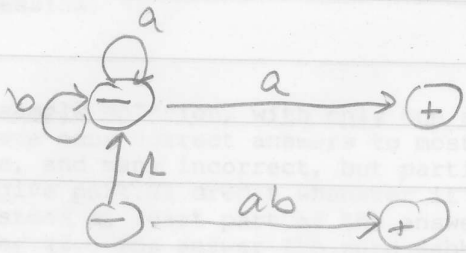
b)



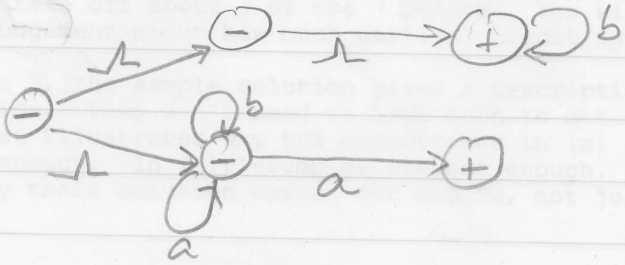
The original TG before the lower (+) was added accepted the language of all words ending in a, which is $(a+b)^*a$. The new language adds ab, but also adds all words ending in ab.

a)





5.



b^*

$(a+b)a^*$

$b^* + (a+b)a^*$