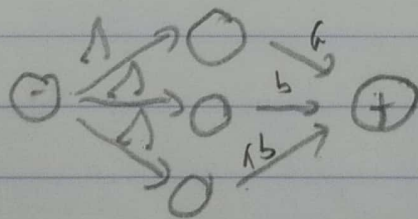
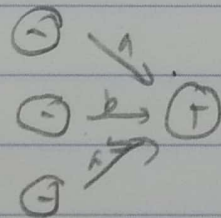


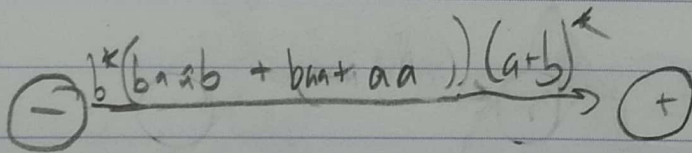
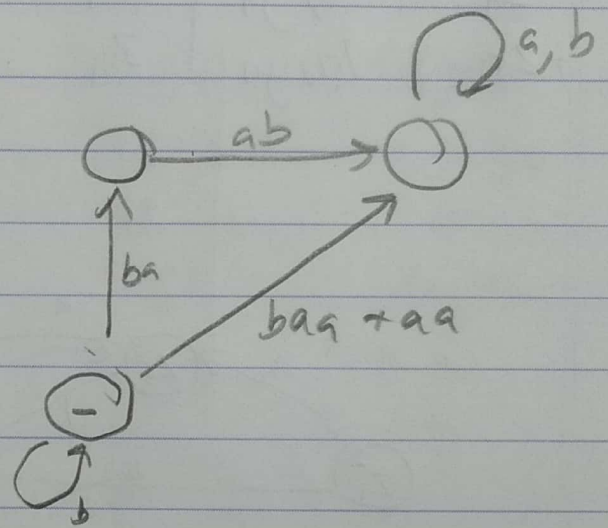
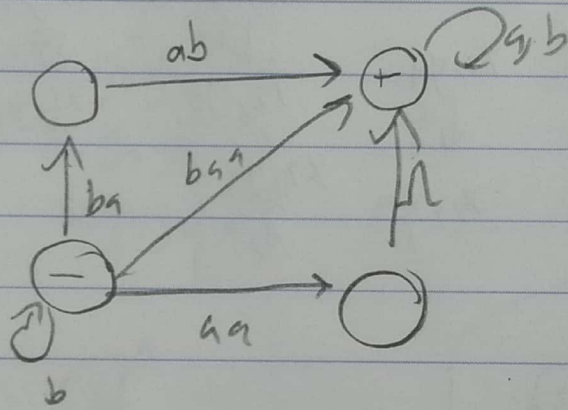
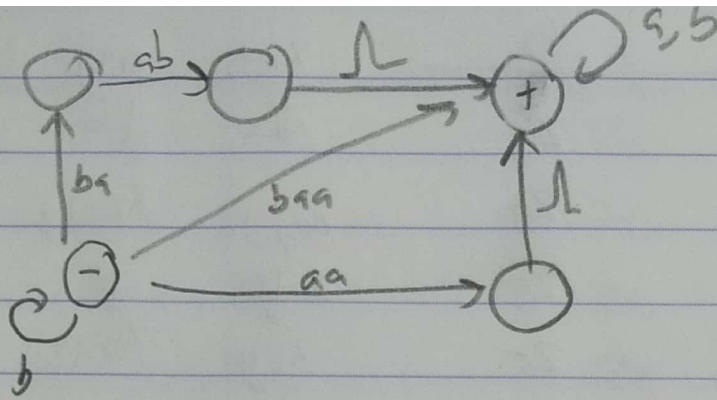
1. Step 1: Transform each TG to a TG with a single start state
 Step 2: Connect one minus state to another minus state by a λ -edge.

Step 1: If a TG has multiple start states, draw a new start state. Details for Step 1: if a TG has multiple start states, draw a new start state. Connect this new state to all the current start states with a λ -edge. The λ -edge(s) goes from the new start state to the current start state. Erase the '-' from the current start states.

EX:



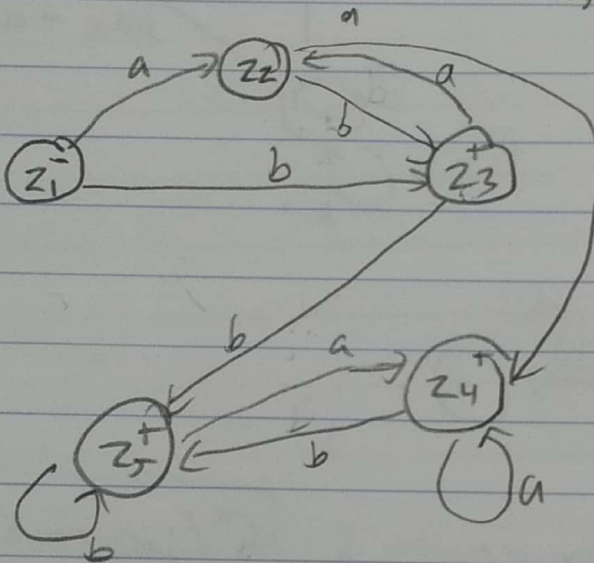
2.



∴ the regular expression is $b^+(baab + baa + aa)(a+b)^*$

Date

| 3, a) | | a | b | x_1, y_1 | x_1, y_1 | |
|--------|------------|-------|-------|----------------|----------------|----------------|
| z_1 | x_1, y_1 | z_2 | z_3 | \downarrow^a | \downarrow^b | |
| z_2 | x_1, y_2 | z_4 | z_3 | x_1, y_2 | x_2, y_3 | x_2, y_3 |
| $+z_3$ | x_2, y_3 | z_2 | z_5 | \downarrow^a | \downarrow^a | \downarrow^b |
| $+z_4$ | x_1, y_4 | z_4 | z_5 | x_1, y_4 | x_1, y_2 | x_2, y_4 |
| $+z_5$ | x_2, y_4 | z_1 | z_5 | x_1, y_2 | x_1, y_4 | x_1, y_4 |
| | | | | \downarrow^b | \downarrow^a | \downarrow^b |
| | | | | x_2, y_4 | x_2, y_3 | x_2, y_4 |
| | | | | \downarrow^a | | |
| | | | | x_1, y_4 | x_2, y_4 | |
| | | | | \downarrow^b | | |



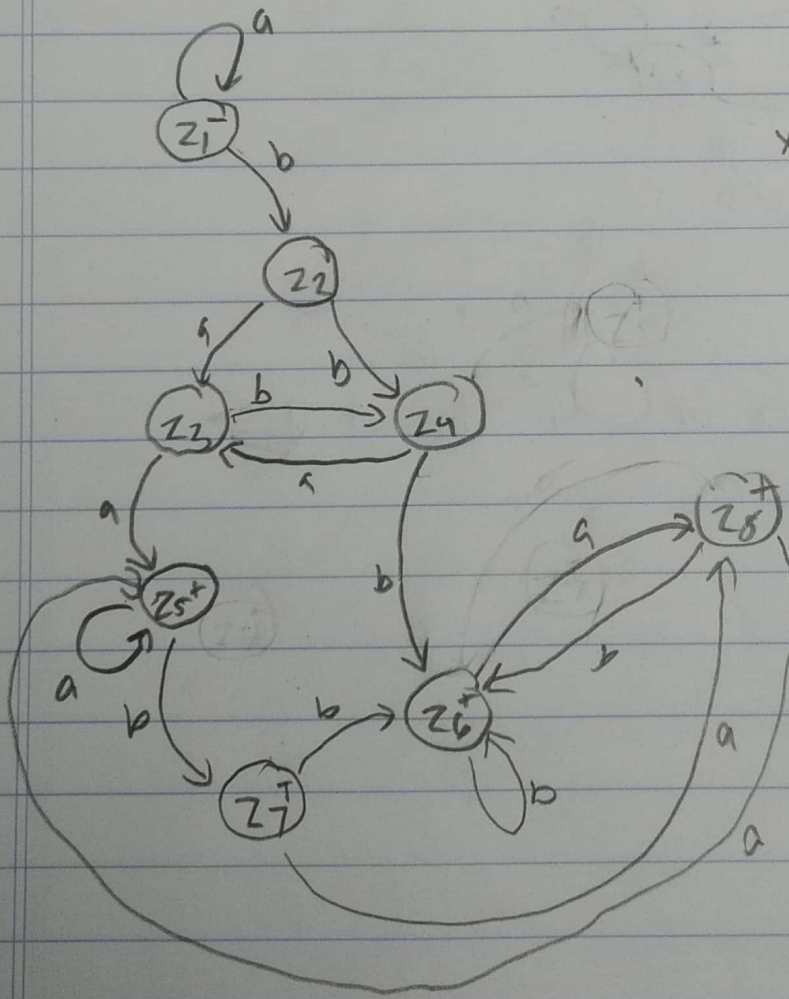
b) bb

c) b

d) aa

4.

| | a | b |
|--------|----------------------|-------|
| Z_1 | x_1 | Z_1 |
| Z_2 | x_2, y_1 | Z_3 |
| Z_3 | x_1, y_2 | Z_5 |
| Z_4 | x_2, y_1, y_3 | Z_3 |
| $+Z_5$ | x_1, y_4 | Z_5 |
| $+Z_6$ | x_2, y_1, y_3, y_4 | Z_8 |
| $+Z_7$ | x_2, y_1, y_4 | Z_8 |
| $+Z_8$ | x_1, y_2, y_4 | Z_5 |



| | | |
|----------|-----------------|----------------------|
| <u>b</u> | x_2, y_1 | x_2, y_1 |
| Z_2 | \downarrow^a | \downarrow^b |
| Z_4 | x_1, y_2 | x_2, y_3, y_1 |
| Z_4 | x_1, y_2 | x_1, y_2 |
| Z_6 | \downarrow^a | \downarrow^b |
| Z_7 | x_1, y_4 | x_2, y_3, y_1 |
| Z_6 | x_2, y_1, y_3 | x_2, y_1, y_3 |
| Z_8 | \downarrow^a | \downarrow^b |
| Z_6 | x_1, y_2 | x_2, y_3, y_4, y_1 |

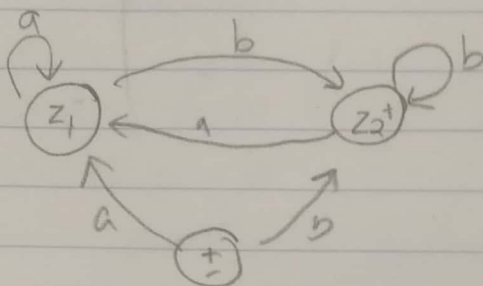
| | |
|----------------------|----------------------|
| x_1, y_4 | x_1, y_4 |
| x_2, y_1, y_3, y_4 | x_2, y_1, y_3, y_4 |
| x_1, y_2, y_4 | x_2, y_3, y_4, y_1 |
| x_2, y_1, y_4 | x_2, y_1, y_4 |
| x_1, y_2, y_4 | x_2, y_3, y_4, y_1 |
| x_1, y_2, y_4 | x_1, y_2, y_4 |
| x_1, y_4 | x_2, y_3, y_4, y_1 |

c) $L_1 L_2$ is the product language that accepts a concatenation of a word in L_1 followed by a word in L_2

b) $L_1 L_2$ contains at least 1 'b' and a double letter. The 'at least one b' cannot be counted towards the double letter. The minimum length is 3. The 1 'b' must come before (but not necessarily directly before) the double letter.

c) aaaaa

| S. | | a | b | x_1, x_2 | x_1, x_2 |
|--------|------------|-------|-------|----------------|----------------|
| Z_1 | x_1 | Z_1 | Z_2 | $\downarrow a$ | $\downarrow b$ |
| $+Z_2$ | x_1, x_2 | Z_1 | Z_2 | x_1 | x_2, x |
| \pm | - | Z_1 | Z_2 | | |



b) The empty string is in L_1^+ but not L_1 .