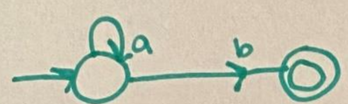
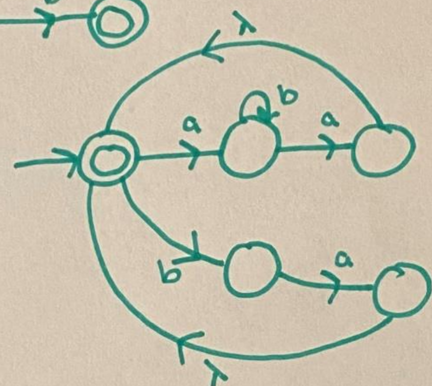
Morgan Baccus
CptS 317
Homework #5

1) $((ab^*a + ba)^* + a^*b)^*$

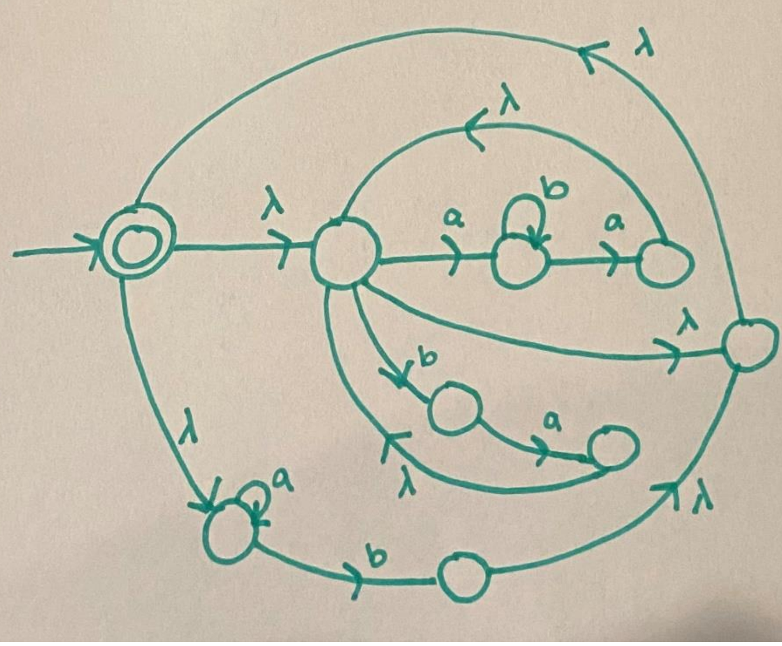
$ab^*a \Rightarrow$ 

$ba \Rightarrow$ 

$a^*b \Rightarrow$ 

$(ab^*a + ba)^* \Rightarrow$ 

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



Morgan Baccus

2)

$P_1 = P_1 a + P_2 b + \epsilon PZ \quad = P_1 b + P2C$

$P_2 = P_2 C + P_1 b$

$P_2 = P_1 b(c)^*$

Then add the expressions together:

$P_1 = P_1 a + P_1 b(c)^* b + \epsilon$

$= P_1(a + bc^* b) + \epsilon \{R = RP + Q \,\&\, R = QP^*\}$

Thus, $P_1 = \epsilon\{a + bc^* b\}^*$ so $(a^* + bc^* b)^*$ &

$P_2 = (a + bc^* b)^* ac^*.$

$\boxed{L(M) = (a + bc^* b)^* ac^*}$

Morgan Baccus

3) We can prove that language L is not regular by using a pumping lemma.

Pumping lemma:

- To prove that L is not a regular language, we follow the concept of proof by contradiction.
- Assume that L is regular.
  Pumping lemma for regular languages there exist a pumping length for L such that:

Conditions:

for any string $s \in L$ where $s = xyz$.

1. $|y| \geq 1$
2. $|xy| \leq p$ where p is pumping length of string.
3. $xy^iz \in L$ where $i \geq 0$

$L = \{ 0^n 1^m : n \geq 1, m \geq 1, n \leq m \}$

we can prove this using pumping lemma.

Step 1: Assume that language L is regular.

Step 2: Consider the strings generated by L:

$n = 1 \quad m = 2 \quad : 0^1 1^2 = 011$

$n = 2 \quad m = 2 \quad : 0^2 1^2 = 0011$

$n = 3 \quad m = 3 : 0^3 1^3 = 000111$

Step 3: Consider a string split into 3 parts

$\underset{x}{0} | \underset{y}{0} | \underset{z}{0\,111}$ where $n = m = p = 3$

3 continued)

Step 4: Now check 3 cases

Case 1:

$|xy| \leq p$

$|00| \leq 3$

$2 \leq 3$

True

Case 2:

$|y| \geq 1$

$|0| \geq 1$

$1 \geq 1$

True

Case 3: check $xy^iz \in L$

Let $i = 2 \Rightarrow 00^2 0111 = 0^4 1^3$

when $n = 4$ $m = 3$

$n \leq m \Rightarrow 4 \nleq 3$

Fail

Solution:

Since case 3 failed, our assumption that language L is regular is false.

$\therefore$ The language L is not regular proved by using pumping lemma.

Morgan Baccus

4) Assume that language L is regular. By pumping lemma there exists an n for L such that any string $s \in L$ when $|s| \geq n$. That is, $s = xyz$ with the following conditions.

1. $|y| > 0$
2. $|xy| \leq n$
3. $xy^iz \in L$, for any $i \geq 0$.

$$x(x^R)x = ab((ab)^R)ab = a(b^n)(b^n)$$
$$aa(b^n)$$

Let $X = ab^n \in L$ so that

If $x = a(b^n)$, $y = b^n$, and $z = a(b^n)$.

However, with $|u| > 0 \geq |(b^n)a| > 0$ being true, the other case is false.

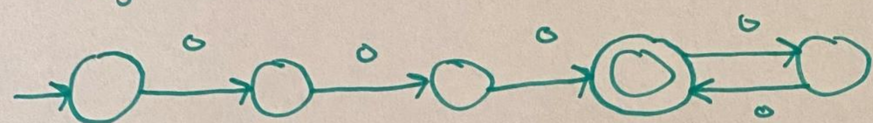
$\boxed{\text{Therefore, } L \text{ is not a regular language.}}$

Morgan Baccus

5)

① Not regular. You would need extra memory to count the 1s and 0s. The last 0s are dependent on the first 1s and 0s.

② Regular. After taking the first four 0s, take anymore it wants. Consider the following DFA:



③ Not regular. Need more memory to create the string. Put all string into the stack with W exempt. Begin with xR from the top of the stack. It will match the element of the top of the stack withe the xR element. POP. compare. If a match, the end will be an empty string.

④ Not regular. Need extra memory. Cannot construct a DFA for this condition.