1-2. Use a general algorithm to design a non-deterministic finite automaton recognizing the language $a^*(b \cup ab)$. After that, use the general algorithm to design a deterministic finite automaton recognizing this same language.
3-4. Use a general algorithm to transform the following finite automaton into the corresponding regular expression.

- This automaton has two states: a state $s_1$ which is a start state and a final state, and another state $s_2$.
- In the state $s_1$, $a$ and $b$ lead to $s_2$, and $c$ leads to $s_1$.
- In the state $s_2$, $a$ leads to $s_2$, and $b$ and $c$ lead to $s_1$.

Let's remove $S_2$:

$$R_{s,F} = R_{s,F} U (R_{s,S_2} R_{S_2} R_{F,S_2})$$
$$= \emptyset U (\emptyset \emptyset a \emptyset) = \emptyset$$

$$R_{s,S_1} = R_{s,S_1} U (R_{s,S_2} R_{S_2} R_{S_2} R_{S_2} S_1)$$
$$= \emptyset U (\emptyset a (a) (a))^*$$

$$R_{s,S_1} = R_{s,S_1} U (R_{S_1 S_2} R_{S_2} R_{S_2} R_{S_2} S_1)$$
$$= c U (a (a) (a)) (buc))$$

$$R_{s,F} = R_{s,F} U (R_{S_1 S_2} R_{S_2} R_{S_2} R_{S_2} F)$$
$$= \emptyset U (c (a) (a)^* \emptyset) = \emptyset$$

continued on previous page
\[ R_{SF} = R_{SF} \cup (R_{S^*}, R_{S^*}, R_{S^*,F}) \]

\[ = \emptyset \cup (\Sigma (\text{cU}(\text{a} \cup \text{b}) (a^*) (b \cup c))^*) \]

\[ = (\text{cU}(\text{a} \cup \text{b}) (a^*) (b \cup c))^* \]
3-4. Use a general algorithm to transform the following finite automaton into the corresponding regular expression.

- This automaton has two states: a state $s_1$ which is a start state and a final state, and another state $s_2$.
- In the state $s_1$, a and b lead to $s_2$, and c leads to $s_1$.
- In the state $s_2$, a leads to $s_2$, and b and c lead to $s_1$.

**Regular Expression:**

\[
L_{s_f} = \varepsilon^* \cup (\varepsilon \: c^* \: \varepsilon) \\
L_{s_{s_2}} = \emptyset \cup (\varepsilon \: c^* \: aub) \\
L_{s_{s_2}^2} = a \cup ((buc) \: c^* \: \varepsilon) \\
L_{s_{s_f}^2} = \emptyset \cup (buc) \: c^* \: \varepsilon \\
L_{s_2} = c^* \cup (c^* \: (aub) \: (au \: (buc) \: c^* \: (aub)) \: ((buc) \: c^*))
\]
5. Let $A$ be a language recognized by the automaton from Problem 3, and let $B$ be the language corresponding to the following automaton:

- This automaton has 3 states, the starting state $r_1$ and two final states $r_2$ and $r_3$.
- When you see $a$, from $r_1$ you move to $r_3$, from $r_2$ you move to $r_3$, and from $r_3$ you move to $r_1$.
- When you see $b$, from $r_1$ you move to $r_3$, from $r_2$ you move to $r_1$, and from $r_3$ you move to $r_2$.
- When you see $c$, from $r_1$ you move to $r_2$, from $r_2$ you move to $r_1$, and from $r_3$ you move back to the same state $r_3$.

Use the algorithm that we learned in class, with pairs of states from two automata as states of the new deterministic automaton, to describe the automata recognizing the union and the intersection of these two languages.
6. (For extra credit) Prove that the language \( \{0^{2n}1^n\} = \{\Lambda, 001, 000011, 000000111, \ldots\} \) is not regular.

Theory: \( L = \{0^{2n}1^n\} \) is not regular. \( \quad \text{Proof by contradiction} \)

Let's assume that \( L \) is regular. Then, by pumping lemma there exists \( p \) such that every word \( w \in L \) whose length is at least \( p \) can be represented as \( xyz \), so that \( \text{len}(y) > 0 \), \( \text{len}(xy) \leq p \), and for every \( i \geq 0 \), \( xy^i z \in L \).

Let's take \( w = 0^{2p}1^p \) so that \( w \in L \), \( \text{len}(w) = 3p \geq p \).

Since \( \text{len}(xy) \leq p \), \( x \) and \( y \) are among the first \( p \) symbols in \( w \), which means that \( y \) can only consist of 0's.

By pumping lemma \( xyyz \) is in \( L \), but when we added an extra \( y \) to \( xyz \), where \( y \) is some number of 0's, we get \( xyyz = 0001, 0000011, \ldots \). Therefore, \( xyyz \) is not in \( L \), which means that \( L \) is not regular.