Task: Apply the general algorithm for transforming the finite automaton into a regular language (i.e., a language described by a regular expression) to the following automaton.

This automaton has two states: x and y; x is the starting state, y is the final state. The only two symbols are 0 and 1.

- From x, 1 leads to y, and 0 to x.
- From y, 1 leads to y, and 0 to x.

Solution. We start with the described automaton:

According to the general algorithm, first we add a new start state ns and a few final state f, and we add jumps:

- from the old start state to the new start state, and
- from each old final state to the new final state.

As a result, we get the following automaton.

Then, we need to eliminate the two intermediate states x and y one by one. We can start with eliminating x or with eliminating y. Let us show what happens in both cases.

First version, when we first eliminate the state y. First, we draw all possible arrows:
Now, to find expressions to place at all these arrows, we will use the general formula

\[ R'_{i,j} = R_{i,j} \cup (R_{i,k} R_{k,k}^* R_{k,j}) , \]

where \( k \) is the state that we are eliminating, i.e., in this case, the state \( k = x \).

By applying this formula, and by using simplification formulas described in the lecture, we get the following results:

\[ R'_{\text{ns},y} = R_{\text{ns},y} \cup (R_{\text{ns},x} R_{x,x}^* R_{x,y}) = \emptyset \cup (\Lambda 0^* 1) = \emptyset \cup 0^* 1 = 0^* 1 ; \]
\[ R'_{\text{ns},f} = R_{\text{ns},f} \cup (R_{\text{ns},x} R_{x,x}^* R_{x,f}) = \emptyset \cup (\Lambda 0^* \emptyset) = \emptyset \cup \emptyset = \emptyset ; \]
\[ R'_{y,y} = R_{y,y} \cup (R_{y,x} R_{x,x}^* R_{x,y}) = 1 \cup (00^* 1) ; \]
\[ R'_{y,f} = R_{y,f} \cup (R_{y,x} R_{x,x}^* R_{x,f}) = \Lambda \cup (00^* \emptyset) = \Lambda \cup \emptyset = \Lambda . \]

Thus, the 3-state a-automaton takes the following form:

Now, all that remains to do is to go from here to the 2-state a-automaton by eliminating the remaining state \( y \):
The final expression is the corresponding expression for $R'_{ns,f}$:

$$R'_{ns,f} = R_{ns,f} \cup (R_{ns,y} R^*_y R_{y,f}) =\emptyset \cup (0^*1(1 \cup (00^*)^* \Lambda) = 0^*1(1 \cup (00^*)^*).$$

The formula on the previous line is a regular expression corresponding to the original automaton.

**First version – answer:** $0^*1(1 \cup (00^*)^*$.

**Second version, when we first eliminate the state $y$.** First, we draw all possible arrows:

Now, to find expressions to place at all these arrows, we will use the general formula

$$R'_{i,j} = R_{i,j} \cup (R_{i,k} R^*_k R_{k,j}),$$

where $k$ is the state that we are eliminating, i.e., in this case, the state $k = y$.

By applying this formula, and by using simplification formulas described in the lecture, we get the following results:

$$R'_{ns,x} = R_{ns,x} \cup (R_{ns,y} R^*_y R_{y,x}) = \Lambda \cup (\emptyset \ldots) = \Lambda \cup \emptyset = \Lambda;$$

$$R'_{ns,f} = R_{ns,f} \cup (R_{ns,y} R^*_y R_{y,f}) = \emptyset \cup (\emptyset \ldots) = \emptyset \cup \emptyset = \emptyset;$$

$$R'_{x,x} = R_{x,x} \cup (R_{x,y} R^*_y R_{y,x}) = 0 \cup (1^*0);$$

$$R'_{s,f} = R_{s,f} \cup (R_{s,y} R^*_y R_{y,f}) = \emptyset \cup (1^* \Lambda) = \emptyset \cup 1^* = 1^*.$$

Thus, the 3-state $a$-automaton takes the following form:
Now, all that remains to do is to go from here to the 2-state a-automaton by eliminating the remaining state \( x \):

The final expression is the corresponding expression for \( R'_{\text{ns},f} \):

\[
R'_{\text{ns},f} = R_{\text{ns},f} \cup (R_{\text{ns},x}R_{x,x}^*R_{x,f}) = \\
\emptyset \cup (\Lambda(0 \cup (11^*0))^*11^*) = \\
\emptyset \cup (0 \cup (11^*0))^*11^* = \\
(0 \cup (11^*0))^*11^*.
\]

The formula in the previous line is also a regular expression corresponding to the original automaton.

**Second version – answer:** \((0 \cup (11^*0))^*11^*\).