Theoretical Computer Science (M21276)

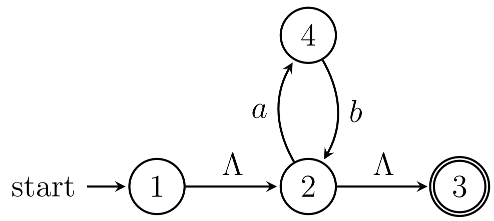
Part A/5: Finite Automata and Regular Languages

(Oct 9-13, 2023)

Question 1. For each of the following regular expressions, construct an NFA using the method described in lecture or your wist.

(i) $(ab)^*$

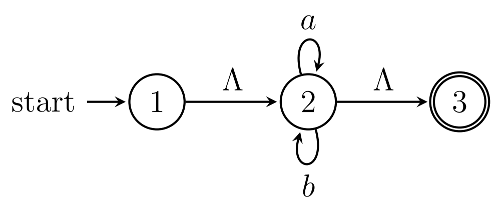
Answer:



(ii) a^*b^*

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

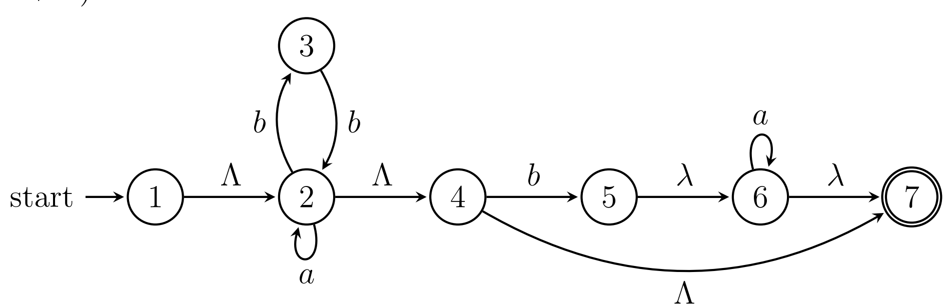
Answer:



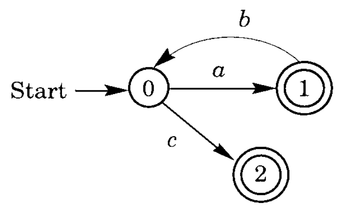
(iv) $a^* + b^*$

Question 2. Find an NFA which accepts the language defined by the regular expression $(a + bb)^*(ba^* + \Lambda)$.

Answer:

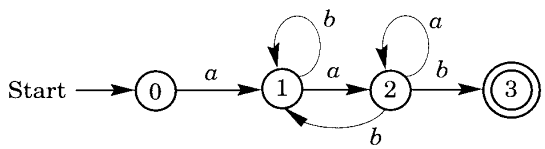


Question 3. Find a regular expression for the language accepted by the following NFA (use the algorithm from the lecture or your wist):



Answer: $(ab)^*(a+c)$

Question 4. Given the following NFA:



Use the algorithm from the lecture to find two regular expressions for the language accepted by the NFA as follows:

(i) Delete state 1 before deleting state 2.

Answer: $ab^*a(a+bb^*a)^*b$

(ii) Delete state 2 before deleting state 1.

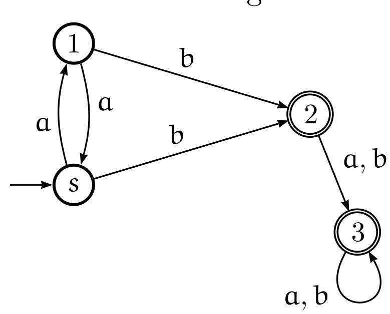
Answer: $a(b+aa^*b)^*aa^*b$

(iii) Prove that the regular expressions obtained in parts (i) and (ii) are equal.

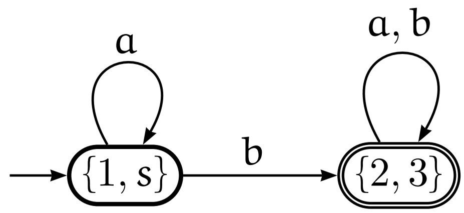
Answer: By using property $(R+E)^* = (R^*E)^*R^*$ we get $(a+bb^*a)^* = (a^*bb^*a)^*a^*$. Similarly using $(R+E)^* = R^*(ER^*)^*$ we can rewrite $(b+aa^*b)^*$ as $b^*(aa^*bb^*)^*$. So, we will be done if we show $ab^*a(a^*bb^*a)^*a^*b = ab^*(aa^*bb^*)^*aa^*b$. Since both expressions have ab^* on the left and a^*b on the right end, it suffices to show that $a(a^*bb^*a)^* = (aa^*bb^*)^*a$. But this is just an instance of the property $R(ER)^* = (RE)^*R$.

Question 5. Given the DFA over the alphabet $\{a, b\}$ with 5 states 0 (initial), 1, 2 (final), 3, 4 (final) and the following transition function: $T(0,a) = T(1,a) = 1$, $T(2,a) = 2$, $T(3,a) = 3$, $T(4,a) = 4$, $T(0,b) = 2$, $T(1,b) = T(3,b) = 4$, $T(2,b) = 3$, $T(4,b) = 2$. Write down the set of equivalent pairs. *Answer:* $\{1, 3\}$

Question 6. Minimise the state in the following DFA:



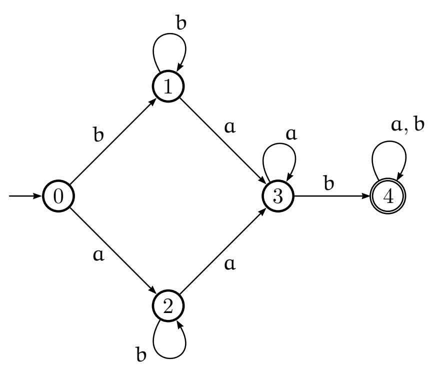
Answer:



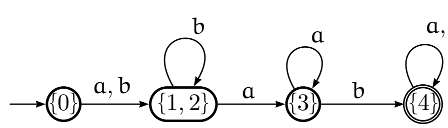
Question 7. For the following DFA over the alphabet $\{a, b\}$ find the minimum-state DFA.
DFA has 5 states: 0 (initial), 1(final), 2, 3 (final), 4 (final) and the following transition function: $T(0, a) = 1$, $T(1, a) = 1$, $T(2, a) = 3$, $T(3, a) = 4$, $T(4, a) = 1$, $T(0, b) = 2$, $T(1, b) = 2$, $T(2, b) = 2$, $T(3, b) = 2$, $T(4, b) = 2$.

Answer: The equivalent pairs are $\{0, 2\}$, $\{1, 3\}$, $\{1, 4\}$, $\{3, 4\}$. Therefore, the states are $\{0, 2\}$, and $\{1, 3, 4\}$, where $\{0, 2\}$ is the start state and $\{1, 3, 4\}$ is the final state. $T_{\min}([0], a) = T_{\min}([1], a) = [1]$, and $T_{\min}([0], b) = T_{\min}([1], b) = [0]$.

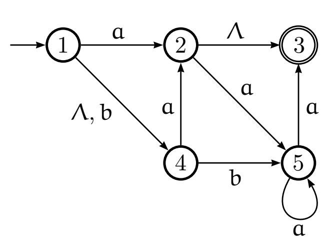
Question 8. Compute the minimum-state DFA for the following DFA.



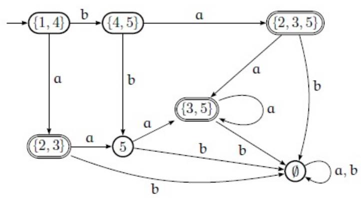
Answer: The set of states is partitioned into the following four equivalence classes: $\{0\}$, $\{1, 2\}$, $\{3\}$, $\{4\}$.



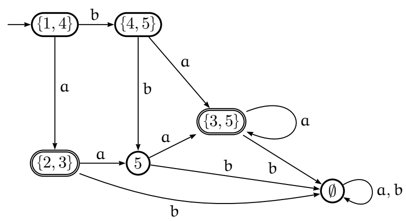
Question 9. Consider the finite automaton below. Construct the minimum-state DFA which accepts the same language. Write down a regular expression that represents the language accepted by your automaton.



Answer: DFA



DFA with the minimum number of states:



Regular expression: $a + (aa + b + bb)a^*a$

Question 10. Transform each NFA from Question 1 into a DFA which will accept the same language, then compute the minimum-state DFA.