Homework 2 Solutions

COMS W3261, Summer B 2021

This homework is due **Monday, 7/12/2021, at 11:59PM EST**. Submit to GradeScope (course code: X3JEX4).

Grading policy reminder: LaTeX is preferred, but neatly typed or handwritten solutions are acceptable. Your TAs may dock points for indecipherable writing. Proofs should be complete; that is, include enough information that a reader can clearly tell that the argument is rigorous.

Remember that the tool `http://madebyevan.com/fsm/` may be useful for drawing finite state machines.

1 Problem 1 (7 points)

Evaluate each of the following regular expressions and write down the language it describes as a set or with a single sentence. (Example: $01^+ = \{w \mid w \text{ consists of a single 0 followed by at least one 1}\}$ or "This regular expression describes the language of strings that consist of a single 0 followed by at least one 1".)

1. (1 point.) Let $\Sigma = \{0, 1\}$. Evaluate $0\Sigma^*1\Sigma^*0$.

   This is the language of all strings over $\Sigma = \{0, 1\}$ that start and end with 0 and contain at least one 1.

2. (1 point.) Evaluate $(01 \cup 0 \cup 1)0^*(1^+)\emptyset$.

   The first part of this expression is concatenated with the empty language $\emptyset = \{\}$, so this is the language $\emptyset$.

Write regular expressions that evaluate to the languages given.

3. (1 point.)

   $\{w \mid w$ consists of a (non-empty) substring of a's and b's of even length, followed by the substring '01'.$\}$

   Note: an earlier version of this question didn't specify that the substring of a's and b's was non-empty, so we will give credit for either interpretation.

   This corresponds to the regular expression $((a \cup b)(a \cup b))^+01$.

4. (1 point.)

   $$\{w \mid w \text{ is a string of 0's with length divisible by } 2, 3, 5, \text{ or } 7.\}$$

   This corresponds to the regular expression $(0^2)^* \cup (0^3)^* \cup (0^5)^* \cup (0^7)^*$.

Evaluate each of the following regular expressions and write down the language it describes as a set or with a single sentence.

5. (1 point.) Let $H = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F\}$. Evaluate $\#HHHHHH$.

   This is the language of all strings that start with # and then contain six symbols from the alphabet H. (Strings in this language correspond to RGB colors expressed in hexadecimal.)

6. (1 point.) Let $P = \{K, Q, R, B, N\}$, $X = \{a, b, c, d, e, f, g, h\}$, $Y = \{1, 2, 3, 4, 5, 6, 7, 8\}$. Evaluate $(P \cup \varepsilon)(\times \cup \varepsilon)XY$.

   This is the language of all strings that consist of (1) a symbol from P (optional), (2) a '$\times$' symbol (optional), (3) a symbol from X, and (4) a symbol from Y, in that order. (Strings in this language correspond to moves in chess notation.)

7. (1 point.) Evaluate the regular expression

   $$\rightarrow (\bigcirc(\underset{0}{\rightarrow} \cup \underset{1}{\rightarrow}))^* \odot .$$

(The alphabet is $\Sigma = \{\rightarrow, \underset{0}{\rightarrow}, \underset{1}{\rightarrow}, \bigcirc, \odot\}$.)

This is the language of all strings that consist of a transition to a start state ($\rightarrow$) followed by 0 or more reject states with a single transition out on 0 or 1 (that is, $\bigcirc \underset{0}{\rightarrow}$ or $\bigcirc \underset{1}{\rightarrow}$), followed by an accept state $\odot$. (Strings in this language correspond to NFAs that accept exactly one string over the alphabet $\{0, 1\}$.)

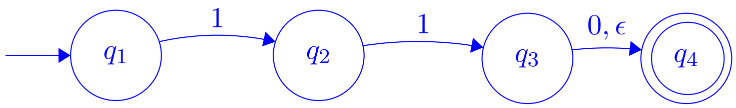
2 Problem 2 (6 points)

1. (6 points). Draw a state diagram for an NFA **with at most 3 states** that recognizes the regular expression

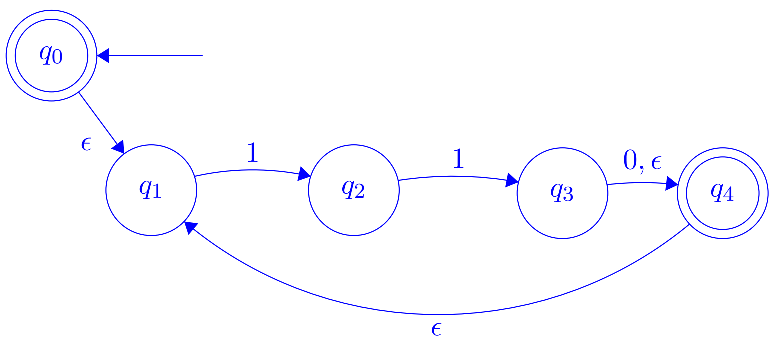
$$(110 \cup 11)^*.$$

Explain in words why your NFA recognizes the language specified.

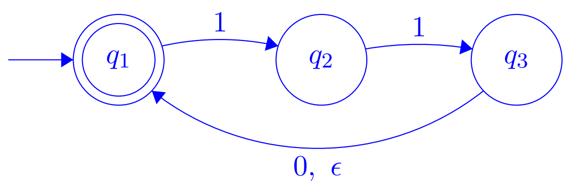
This regular expression evaluates to the language of all strings that consist of concatenated '110' and '11' substrings. First, we observe that the regular expression $(110 \cup 11)$ is recognized by the following NFA:



Using the procedure from the proof that the regular languages are closed under star, we create a 5-state NFA that recognizes $(110 \cup 11)^*$:



Finally, we simplify to the following equivalent 3-state NFA:



This NFA accepts a string if and only if some branch of computation is in state q_1 once all input symbols are read. For every string consisting of n concatenated 110 or 11 substrings, there is clearly at least one branch of computation that goes around the loop of our NFA n times and accepts. Conversely, if some branch of computation accepts after traversing the loop $n \geq 0$ times, this computation must correspond to n concatenated substrings from the set $\{110, 11\}$.

3 Problem 3 (6 points)

1. (6 points.) Given a language A, define the language $\text{pre}(A)$ as follows:

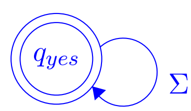
$$\text{pre}(A) := \{xy \mid x \in A\}.$$

Prove that the class of regular languages is closed under $\text{pre}(\cdot)$.

One way to prove this is to follow the pattern of our proofs that the regular languages are closed under regular operations. Specifically, we'll show how to modify the state diagram of any NFA that recognizes the language A in such a way that the new state diagram represents an NFA that recognizes the language $\text{pre}(A)$.

Consider any regular language A and let N be an NFA that recognizes it. We'd like to create an NFA N' that accepts $\text{pre}(A)$: in other words, N' should accept if any prefix of the input string is contained in A. This is equivalent to accepting if any branch of computation on N ever reaches an accept state, even at an intermediate step of computation.

To accomplish this, we add a new accept state q_{yes} that looks as follows to N. (Here, the Σ symbol is shorthand for transitions that move us from q_{yes} to q_{yes} on any input symbol in the alphabet Σ.)



We then add ε-arrows from every accept state in N to q_{yes}. This completes N' and ensures that on any string xy such that $x \in A$, some branch of our computation ends up in q_{yes}. This branch survives until the end of the computation, ensuring that we accept.

(Alternatively, we might observe that $pre(A)$ is equivalent to the language A concatenated with the language Σ^* of all strings over Σ. Because we can draw a DFA that accepts Σ^*, Σ^* is regular. As A is regular by assumption, $pre(A) = A \circ \Sigma^*$ is regular by the closure of regular languages under concatenation.)

4 Problem 4 (1 point)

1. What is one thing the instructor or course staff could do better to make the material or expectations clearer or more convenient for you?

   (Any coherent response.)

2. Adjusting for the compressed timeframe of this course (two weeks in one), how are you finding the difficulty so far in relation to other CUCS courses? (For example: 'much easier', 'somewhat easier', 'about the same', 'somewhat harder', 'much harder'.)

   (Any coherent response.)

3. (Optional) Any other thoughts? Thank you!