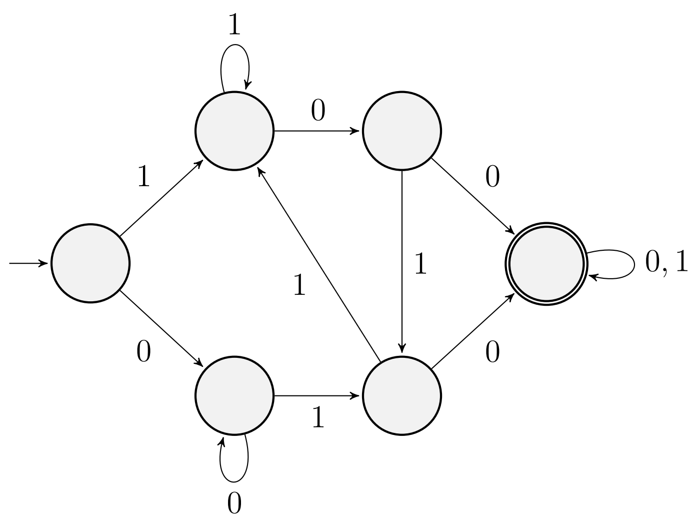
Assignment 1

Naman Mishra

16 January, 2024

Problem 1.1. Give a DFA for the language of all strings over the alphabet $\{0, 1\}$ which contain an occurrence of 010 or 100 as a contiguous substring.

Solution.

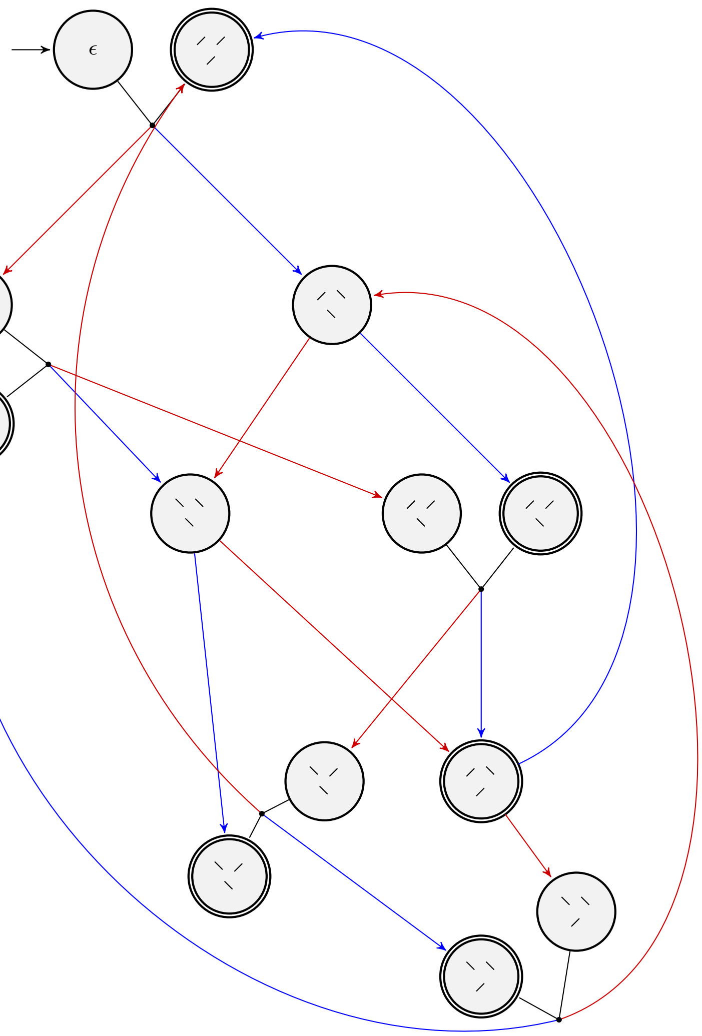


Problem 1.2.

Solution. We take the following liberties while drawing the DFA.

- If two states q and q' are such that $\delta(q, a) = \delta(q', a)$ for all $a \in \{A, B\}$, where δ is the transition function, we join them to a "junction" in between, and draw a single path from the junction instead of one from each state.

- Instead of labelling each edge with the input symbol, we color the edges red for input A and blue for input B.



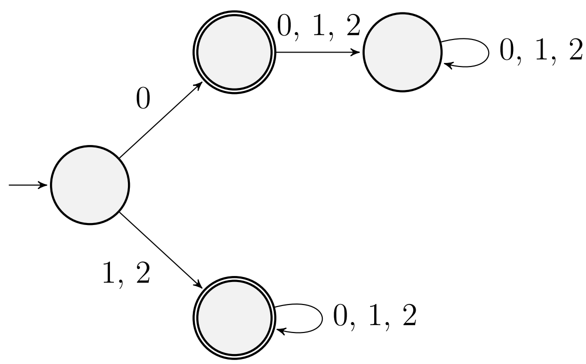
Problem 1.3. Show that the set of strings in $\{0, 1, 2\}^*$ which are base 3 representations of even numbers, is regular.

Proof. Let $s \in \{0, 1, 2\}^*$ be a non-empty string. We zero-index the string from right to left. Then s encodes the number $n = \sum_{i=0}^{n} s_i 3^i$ in base 3.

$$
\begin{aligned}
n &= \sum_{i=0}^{n} s_i 3^i \\
n &\equiv \sum_{i=0}^{n} s_i 1^i \pmod 2 \\
n &\equiv \#_1(s) \pmod 2
\end{aligned}
$$

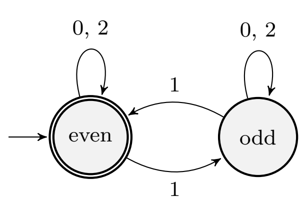
where $\#_1(s)$ is the number of 1's in s. Thus n is even iff s contains an even number of 1's.

We first give a DFA which accepts Z, the set of valid ternary representations of all natural numbers, *i.e.*, the set of all non-empty strings over A which either start with 1 or 2, or are exactly "0".



This gives us that Z is regular.

Let E be the set of all strings over $\{0, 1, 2\}$ which are base 3 representations of even numbers, possibly with leading zeroes and with the empty string counting as a representation of 0. This is a regular language, as it is accepted by the following DFA.



Since regular languages are closed under intersection, E intersected with Z is also regular. However, this is precisely the set of all valid ternary representations of even numbers as required in the problem. $\square$

Problem 1.4. Let $\mathcal{A} = (Q, s, \delta, F)$ be a DFA over alphabet A. Give a formal proof that for any strings $x, y \in A^*$:

$$\widehat{\delta}(s, xy) = \widehat{\delta}(\widehat{\delta}(s, x), y)$$

Proof. We will prove this by induction on the length of y. For $\#y = 0 \iff y = \epsilon$, we have

$$\begin{aligned}\widehat{\delta}(s, x\epsilon) &= \widehat{\delta}(s, x) \\ &= \widehat{\delta}(\widehat{\delta}(s, x), \epsilon)\end{aligned}$$

by definition of $\widehat{\delta}$.

Suppose the statement holds for all y of length $\#y = n \geq 0$. Let y be a string of length $\#y = n + 1$. Then $y = y'a$ for some y' of length n and $a \in A$.

$$\begin{aligned}\widehat{\delta}(s, xy) &= \widehat{\delta}(s, xy'a) \\ &= \delta(\widehat{\delta}(s, xy'), a) && \text{(definition)} \\ &= \delta(\widehat{\delta}(\widehat{\delta}(s, x), y'), a) && \text{(induction hypothesis)} \\ &= \widehat{\delta}(\widehat{\delta}(s, x), y'a) && \text{(definition)} \\ &= \widehat{\delta}(\widehat{\delta}(s, x), y).\end{aligned}$$

Thus by induction, the statement holds for all $y \in A^*$. Since x was arbitrary, the statement holds for all $x, y \in A^*$. $\square$

Problem 1.5. Consider the language of nested C-style comments. The aphabet comprises characters "`/`", "`*`", and "`c`" (the latter symbol representing any ASCII character apart from "`/`" and "`*`"). The language allows all well-nested and complete comments. Thus strings like "`cc/*ccc*/c`" and "`cc/*cc/*ccc*/c*/ccc`" are in the language, but not "`cc/*c/*cc*/cc`". Is this language regular? Justify your answer.

Proof. Suppose this language L is regular. Let k be a pumping length for

L. That is, let $k \in \mathbb{N}$ be such that for any word $t \in L$ of the form xyz with $|y| \geq k$, there exist strings u, v and w with $y = uvw$, $v \neq \epsilon$, and $xuv^iwz \in L$ for each $i \geq 0$.

Let $t = xyz$ where $x = (\texttt{/*})^k$, $y = (\texttt{*/})^k$, and $z = \epsilon$. Then $t \in L$ since all comments are well-nested and closed. Let u, v and w be strings following the above statement for this y.

Since the first $2k$ characters of $xuwz$ constitute k comment opening delimiters, but the rest of the string has less than $2k$ characters, it cannot contain k comment closing delimiters. Thus $xuwz \notin L$, a contradiction.

Therefore L can not be regular. $\square$

Problem 1.6. For a set of natural numbers X, define *binary*(X) to be the set of binary representations of numbers in X. Similarly define *unary*(X) to be the set of "unary" representations of numbers in X: *unary*$(X) = \{1^n \mid n \in X\}$. Thus for $X = \{2, 3, 6\}$, *binary*$(X) = \{10, 11, 110\}$ and *unary*$(X) = \{11, 111, 111111\}$. Consider the two propositions below:

(a) For all X, if *binary*(X) is regular then so is *unary*(X).

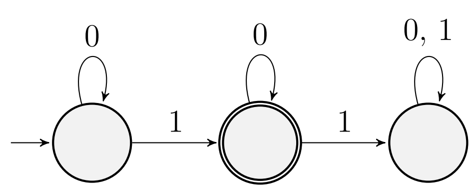
(b) For all X, if *unary*(X) is regular then so is *binary*(X).

One of the statements above is true and the other is false. Which is which? Justify your answer.

The second statement is true and the first is false. We first prove the falsity of the first statement. One crucial observation for both proofs is that

$$lengths(unary(X)) = \{|1^n| : n \in X\} = X.$$

Proof. Let $X = \{2^n \mid n \in \mathbb{N}\}$. Then *binary*$(X) = \{10^n \mid n \in \mathbb{N}\}$ is regular. A DFA for this language is



However, $unary(X) = \left\{1^{2^n} \mid n \in \mathbb{N}\right\}$ is not regular as its lengths, that is X, is not ultimately periodic. This is because for any alleged period $p \in \mathbb{N}^+$ and any $N \in \mathbb{N}$, we have $2^{N+p} \in X$ but $p + 2^{N+p} \notin X$ since $2^{N+p} < p + 2^{N+p} < 2^{N+p+1}$ (note also that $2^{N+p} > N$). $\square$

We now prove the truth of the second statement.

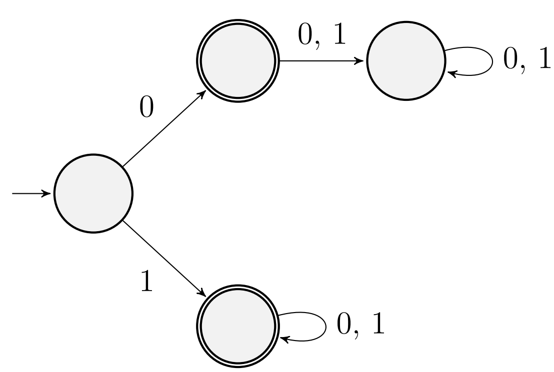
Proof. Let X be such that $unary(X)$ is regular. Then X is ultimately periodic. Let $N, p \in \mathbb{N}^+$ be such that for all $n \geq N$, $n \in X \iff n + p \in X$.

Then $X = E \cup \{n \in \mathbb{N} \mid n \geq N, n \bmod p \in K\}$ for some finite subset $E \subseteq \{0, 1, \ldots, N-1\}$ and some $K \subseteq \{0, 1, \ldots, p-1\}$. We write X as $E \cup (M \cap B)$, where

$$M = \{n \in \mathbb{N} \mid n \bmod p \in K\}$$
$$B = \{n \in \mathbb{N} \mid n \geq N\}$$

Since there exists a unique binary representation of each natural number (without leading zeroes), $binary(X) = binary(E) \cup (binary(M) \cap binary(B))$. We show that each of these three sets is regular by constructing DFAs over the alphabet $A = \{0, 1\}$ that accept them.

We first give a DFA which accepts Z, the set of valid binary representations of all natural numbers, *i.e.*, the set of all non-empty strings over A which either start with 1 or are exactly "0".



This gives us that Z is regular.

We define $binary^\infty(S)$ to be the set of all strings which are any binary representation of some number in S, possibly with leading zeroes and with the empty string counting as a representation of 0. Note that though one natural number may have multiple binary representations, each binary representation is a representation of a unique natural number.

Then $binary(S) = binary^\infty(S) \cap Z$ for all $S \subseteq \mathbb{N}$. Thus it suffices to show that $binary^\infty(E)$, $binary^\infty(M)$ and $binary^\infty(B)$ are regular, since regular languages are closed under intersection.

Let

$$\begin{aligned} Q &= \{0, 1, \ldots, N\} \\ \delta(q, a) &= \begin{cases} \min\{2q, N\} & \text{if } a = 0, \\ \min\{2q+1, N\} & \text{if } a = 1. \end{cases} \\ \mathcal{E} &= (Q, 0, \delta, E) \\ \mathcal{B} &= (Q, 0, \delta, \{N\}) \end{aligned}$$

Here, each state $q \in \{0, 1, \ldots, N-1\}$ corresponds to the property that the string read so far represents q, whereas state N corresponds to the property that it represents some number greater than or equal to N. δ preserves each property and thus $\mathcal{E}$ and $\mathcal{B}$ accept $binary^\infty(E)$ and $binary^\infty(B)$ respectively. This gives regularity of $binary(E)$ and $binary(B)$.

Finally, $binary^\infty(M)$ is regular since it is accepted by

$$\mathcal{M} = (\{0, \ldots, p-1\}, 0, \delta', K),$$

where

$$\begin{aligned} \delta'(q, 0) &= 2q \bmod p \\ \delta'(q, 1) &= (2q+1) \bmod p \end{aligned}$$

This gives regularity of $binary(M)$.

Since regular languages are closed under union and intersection, $binary(X)$ is also regular. $\square$