COMP 598

Talise Wang 260829722

May 12, 2020

Question 1

We have $f : \Sigma \to M$. Create a monoid homomorphism $f^* : \Sigma^* \to M$.
Define Σ_i be the set containing element of length i such that $\Sigma^* = \Sigma_0 \cup \Sigma 1 \cup \cdots$
For f^* on Σ_0: $f^*(\epsilon) = e$ where e is the identity element of M.
For f^* on Σ_1: $f^*(c) = f(c)$
For $i \geq 2$, we will define them recursively. given $wx \in \Sigma_i$ where $w \in \Sigma_{i-1}, x \in \Sigma$, $f^*(wx) = f^*(w)f(x)$
Now let's prove that f^* is a monoid homomorphism:
The identity element is explicitly perserved.
Let $*$ be the binary operation on M and $\cdot$ for Σ^*, let $a_i, b_i \in \Sigma$. WTS $f^*(x \cdot y) = f^*(x) * f^*(y)$

$$
\begin{aligned}
f^*(x) * f^*(y) &= (f(a_1) * f(a_2) * \cdots * f(a_n)) * (f(b_1) * f(b_2) * \cdots * f(b_n)) \\
&= f(a_1) * f(a_2) * \cdots * f(a_n) * f(b_1) * f(b_2) * \cdots * f(b_n) \qquad \text{by associativity} \\
&= f^*(a_1 \cdot \ldots \cdot a_n \cdot b_1 \cdot \ldots \cdot b_n) \\
&= f^*(x \cdot y)
\end{aligned}
$$

We can prove this homomorphism is unique by induction.
Since any other function f' which agree with f^* on Σ_i will agree on Σ_{i+1} since $f'(xy) = f'(x)f(y) = f*(x)f(y) = f^*(xy)$. So it's unique.

Question 2

$\to$ Let Σ be the alphabet. L is recognized by a DFA $A = (S, s_0, \sigma : S \times \Sigma \to S, F)$.

Define a transition monoid $M = (\{\delta_x \text{ where } x \in \Sigma^*\}, \circ, Id_s)$ with $\delta_x(s) = \delta^*(s, x)$ and Id_s means the identity function. So M is a set of functions.
To prove M is monoid.
First, define functions $x, y \in M$, $x \circ y = y \circ x$. Since $\forall x, y, z \in M, x \circ (y \circ z) = x \circ (z \circ y) = (x \circ y) \circ z = (y \circ x) \circ z = (y \circ z) \circ x = z \circ (x \circ y) = z \circ (y \circ x)$.
And $\forall x \in M$,$x \circ Id_s = Id_s \circ x = x$.
So proved.
Define the homomorphism $f : \Sigma^* \to M$, $f(x) = \delta_x$ and $F' = \{\delta_x | x \in \Sigma^*, \delta_x(s_0) \in F\}$. So that this language can be recongnized by this (M, F', f), so $L(M, F', f) = \{w \in \Sigma^* | f(w) \in F\}$.
$\leftarrow$
Since $L \subset \Sigma$ is recognized by a finite monoid $(M, \circ, 1)$, then there exists a homomorphisim $f : \Sigma^* \to M$and a subset $F \subset M$ such that $L = f^{-1}(F)$. This means $\forall\ w \in \Sigma^*$, $\exists\ s \in M$ such that $f(w) = s$ holds and $\forall\ w \in L$, $\exists\ s \in F$ such that $f(w) = s$ holds.
Define a DFA $A = (M, 1, \sigma, F)$ where $\delta(m, a) = m \circ f(a)$. To formalizing acceptance: $\delta^* : S \times \Sigma^* \to S$. Base case: $\sigma^*(s', \epsilon) = s'$. Then by induction on length of $w \in \Sigma^*$, $\delta^*(s', w \circ a) = \delta(\underbrace{\delta^*(s', w)}_{s}, a)$ holds for $\forall\ a \in \Sigma$.

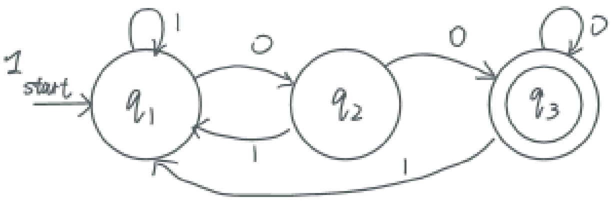
So we can get $L(A) = \{w \in \Sigma * | \delta^*(1, w) \in F\}$. This means L can be recognized by a DFA A.

Question 3

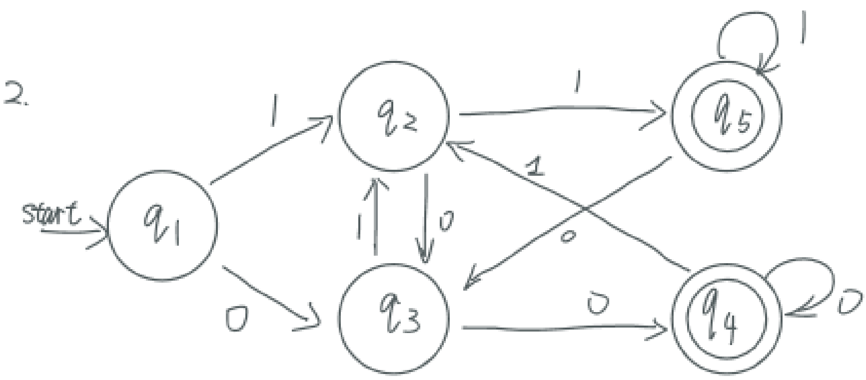
Suppose h is not inflationary, which means $V = \{x | x > h(x)\}$ and $V \neq \emptyset$. W is well-founded so that V has minimal element such that $v_0 > h(v_0)$. Since $\forall y < v_0$,$y \leq h(y)$, we have $h(v_0) \leq h(h(v_0))$. However, since h is strictly monotone, we cam do $h(v_0) \geq h(h(v_0))$ from $v_0 \geq h(v_0)$. So here comes to an contradiction!
So proved! h must be an inflationary

Question 4

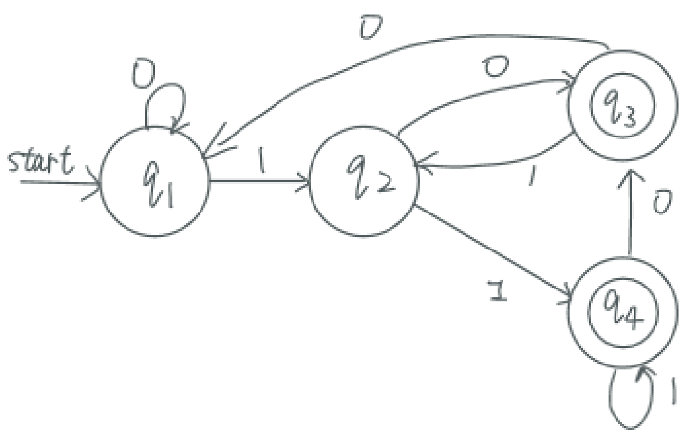
1



2.



3.



Question 5

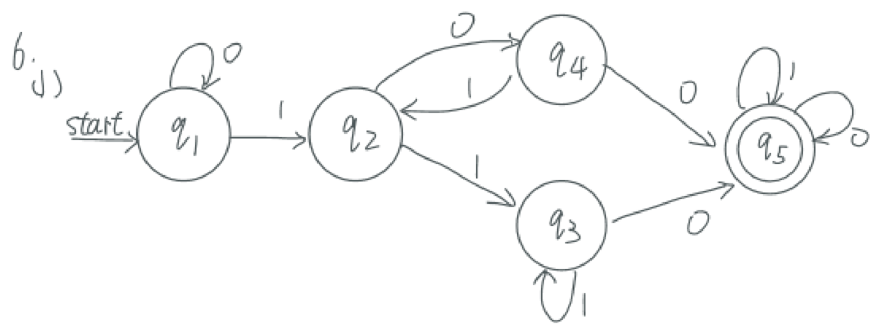
Assume (S, s_0, F, δ) recognizes L. We'll construct an NFA (Q, Q_0, F, Δ) for middle(L).

Let's define state space $Q = \overbrace{S}^{\text{track } v} \times \underbrace{S}_{\substack{\text{guess where DFA will} \\ \text{be when } u \text{ ends}}} \times \overbrace{S}^{\substack{\text{guess where DFA will} \\ \text{be when } v \text{ ends}}} \times \underbrace{2^S}_{\text{track } u} \times \underbrace{2^S}_{\text{track } w}$

Then $Q_0 = \{(s, s, t, \{s_0\}, \{t\})\}$, $F' = \{(t, s, t, X, Y) | s \in X, Y \cap F \neq \emptyset\}$.
$\Delta = ((s, s_{const}, t, X, Y), a) = \{(s', s_{const}, t, X', Y') | \delta(s, a) = s',\ X', Y'$ is the set of states of states DFA can reach from X & Y $\}$.
So it's regular since we can construct a NFA for middle(L).

Question 6

6.1



6.2

Suppose for contradiction that we can find a correct DFA that has 4 states. By the Pigeonhole Principle, if we choose 5 strings over Σ, then at least two of those strings must end at the same state s.
Let wi and wj be these 2 strings that satisfies the above.
For any string x, if wi and wj end at s, then wix and wjx must also end at the same state s' and hence must both be accepted or rejected by the DFA.

And we will show for each possible pair, there exists some x which makes above statement false. That will raise a contradiction, so we cann't find a DFA that recongnize this language, which means any DFA that recongnize this language must have at least 5 states.
So we will do above by choosing these five strings in Σ:
The five strings will be:
$W0 = \epsilon$
$W1 = 110$
$W2 = 010$
$W3 = 111$
$W4 = 101$
Then we will do pair matching one by one.
Pair #1 Pick $x = \epsilon$
then $W_0x = \epsilon$ reject $W_1x = 110$ accept
Pair #2 Choose $x = 0$
$W_0x = 0$ reject $W_2x = 0100$ accept
Pair #3 Choose $x = 10$
$W_0x = 10$ reject $W_3x = 11110$ accept
Pair #4 Choose $x = 10$
$W_0x = 10$ reject $W_4x = 10110$ accept
Pair #5 Choose $x = \epsilon$
$W_1x = 110$ accept $W_2x = 010$ reject
Pair #6 Choose $x = \epsilon$
$W_1x = 110$ accept $W_3x = 111$ reject
Pair #7 Choose $x = \epsilon$
$W_1x = 110$ accept $W_4x = 101$ reject
Pair #8 Choose $x = 10$
$W_2x = 01010$ reject $W_3x = 11110$ accept
Pair #9 Choose $x = 10$
$W_2x = 01010$ reject $W_4x = 10110$ accept
Pair #10 Choose $x = 0$
$W_3x = 1110$ accept $W_4x = 1010$ reject

Question 7

Assume (S, s_0, F, δ) is a DFA that recognizes L. Since a boolean matrix is finite and thus can be used in a NFA. We will this to know if $\exists y$.

To bulid a NFA (Q, Q_0, F, Δ):
First, Define $M_{n\times n}$ as a boolean matrix where each entry M_{ij} in M count the number of steps we need to use to get from s_i to s_j.
Assume it takes $|w|$ number of char to get to the s_j, then the steps number in M_{ij} will be $2^{|w|}$.
So at the end, we can know from M by seeing the entries if there is a y such that $xy \in L$ & $|y| = 2^{|x|}$.
Then define State space $Q = S \times M$ where S tracks where the specific state is when the machine is reading x.
$Q_0 = \{(s_0, M)\}$
$F' = \{(s_i, M)|M_{ij} = 1 \text{ for } s_j \in F\}$
$\Delta((s, M), a) = (\delta(s, a), M^2)$
So we constructed a NFA successfully! Clearly, it's regular.

Question 8

8.1

Pick $p > 0$ arbitrary, we choose $w = a^{2p}b^p$ such that $|w| > p$ always hold.
Since $|xy| \leq p$ & $|y| > 0$ hold, y can only consist exclusively for a's, so $y = a^k$ for some k with $1 \leq k \leq p$.
Pick $i = 0$, then $xy^0z = xz = a^{2p-k}b^p$
If $xy^0z \in \{a^{2n}b^n\}$, then $2p - k = 2p$ and we will get $k = 0$, but $k \leq 1$, so impossible, so $xy^0z \notin \{a^{2n}b^n\}$. So by pumping lemma, it is not regular.

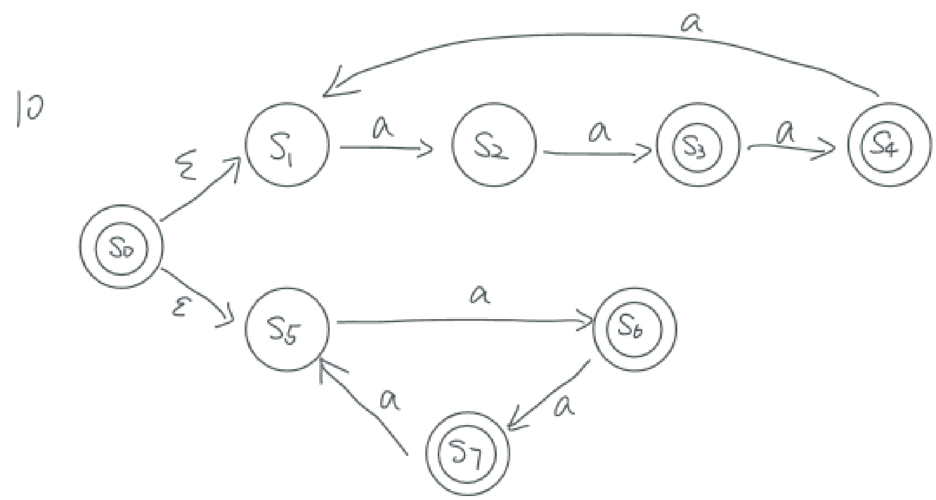
8.2

Let $L = \{x \in \{a, b, c\}^* || x| \text{ is a square}\}$.
Pick $p > 0$ arbitrary, we choose $w = a^{(p+1)^2}$. So $w \in L$ and then $|w| > p$ always hold.
Since $|xy| <= p$ & $|y| > 0$ hold, y can only consist exclusively for a's, so $y = a^k$ for some k with $1 \leq k \leq p$.
Pick $i = 2$, then $xy^iz = xy^2z = a^{(p+1)^2+k}$, so $|xy^2z| = (p + 1)^2 + k$
Obviously, $(p+1)^2+k > (p+1)^2$. Since $k \leq p$, so $(p+1)^2+k \leq (p+1)^2+p < (p+1)^2+2p+3 = (p+2)^2$. So $(p+1)^2 < (p+1)^2+k < (p+2)^2$ and obviously $xy^2z \notin L$.
So by pumping lemma, it is not regular.

Question 9

Counterexample: $\Sigma = \{a, b, c\}$,let $L = \{a^*cb^*\}$. Since $c \in \Sigma$, c is regex. Clearly, a^*,b^* are regex. So $\{a^*cb^*\}$ can be described as regex which means L is regular.
By thm, a language is regular iff it can be described as regex.
So if outer(L)$\cap\{a^*b^*\}$ is not regular, then outer(L) can not be regular by closure properties of regular languages.(since $\{a^*b^*\}$ is regular)
Since $L = \{a^*cb^*\}$ & there no c in $\{a^*b^*\}$.
So for uw $\in$outer(L)$\cap\{a^*b^*\}$, there's not c in u or w. So c can only be in v.
Since $L = \{a^*cb^*\}$ and c can only be in v. So $u = a^*$ & $w = b^*$ & $v = a^*cb^*$ with $|u| = |v| = |w|$. So $u = a^n$,$w = b^n$ with the same n.($n \geq 1$)
So for outer(L)$\cap\{a^*b^*\}$, we'll get outer(L)$\cap\{a^*b^*\} = \{a^nb^n, n \geq 1\}$.
Then we will use pumping lemma to prove $\{a^nb^n, n \geq 1\}$ is not regular.
Pick $p > 0$ arbitrary, we choose $w = a^pb^p$, so clearly $|w| = 2p > p$ holds.
Since $|xy| \leq p$ & $|y| > 0$ hold, y can only consist exclusively for a's, so $y = a^k$ for some k with $1 \leq k \leq p$.
Pick $i = 2$, then $xy^iz = xy^2z = a^{p+k}b^p$. Since $k > 0$, so $p + k \neq p$.
So $xy^2z \notin \{a^nb^n, n \geq 1\}$. So $\{a^nb^n, n \geq 1\}$ is not regular.
So proved! outer(L) is not regular!

Question 10



Define $\Sigma = \{a\}$
And the shortest string that it rejects has length 9 which is bigger than 8(states numbers).