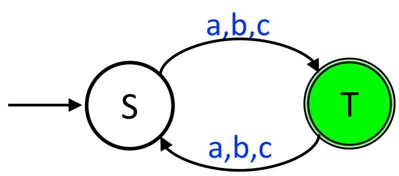
10 (100 PTS.) Prove Regular

Let Σ be finite alphabet. A ***code*** is a mapping $f : \Sigma \to \{0, 1\}^+$. For example, if $\Sigma = \{a, b, c\}$, a code f might be $f(a) = 101$, $f(b) = 01100$, and $f(c) = 10$. (To simplify things, we assume for any $a \neq b$, we have $f(a) \neq f(b)$.)

For a string $w_1 w_2 \cdots w_m \in \Sigma^*$, we define $f(w) = f(w_1)f(w_2)\cdots f(w_m)$. In the above code,

$$f(abcba) = 101 \bullet 01100 \bullet 10 \bullet 01100 \bullet 101. = 1010110010011001 01.$$

10.A. (10 PTS.) Let L be the language of the following DFA M. What language does L represent ?



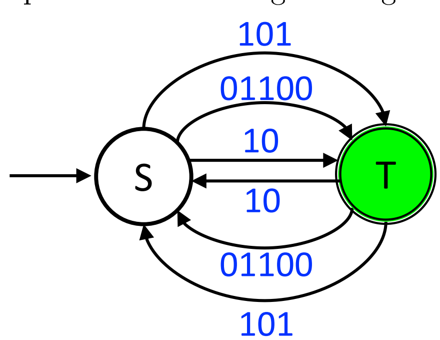
Solution:

L is the language of all odd length stings over $\{a, b, c\}$.

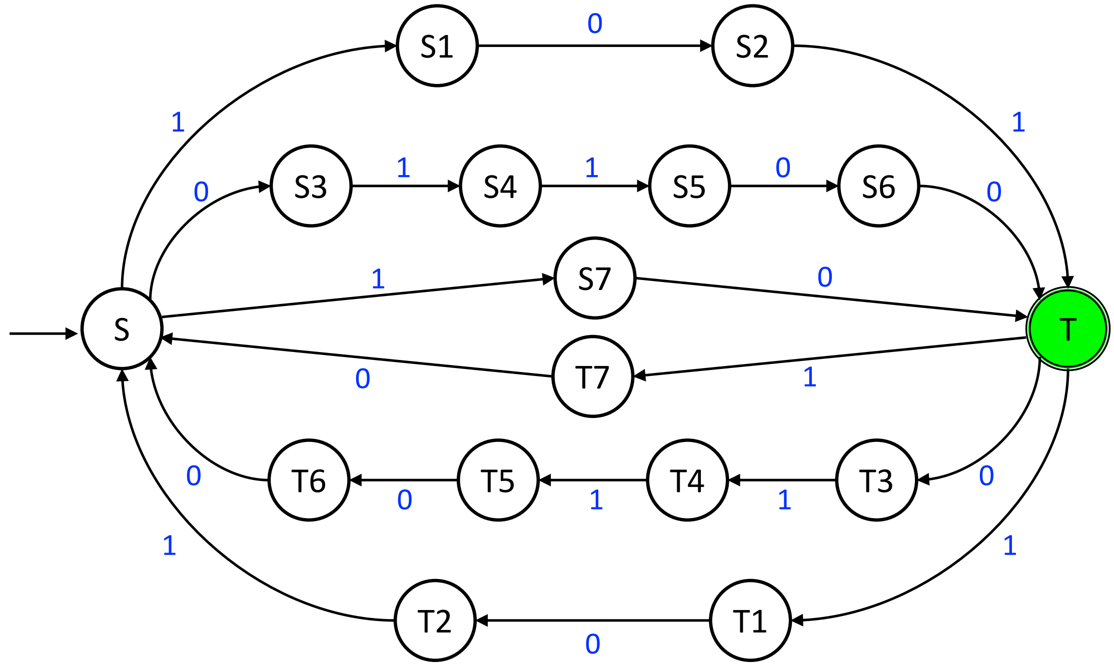
10.B. (20 PTS.) Working on the DFA M from **10.A.** construct an NFA for the language $f(L)$. Here, $f(L) = \{f(w) \mid w \in L\}$ is the *code language* where f is code from the above example.

Solution:

First lets do direct replacement. Making the edges correspond to strings.



Next, introduce middle states, so that every transition is a single character.



10.C. (70 PTS.) Let $L \subseteq \Sigma^*$ be a arbitrary regular language. Prove that the coded language $f(L) = \{f(w) \mid w \in L\}$ is regular.

Specifically, given a DFA $M = (Q, \Sigma, \delta, s, A)$ for L, describe how to build an NFA N for $f(L)$. Then, prove the correctness of your construction, i.e., the language of the constructed NFA is indeed the desired language $f(L)$. Your construction and proof should be for any arbitrary code f and not just the code in the example above. (30 points for a correct construction, and 40 points for a correct proof of correctness.)

Solution:

We follow the same scheme as (B). Let N be an NFA that is initially a copy of M with the same start and accept states but without any transitions.

We now loop over all states $q \in Q$, and all characters $a \in \Sigma$. For such a pair q, a, let $w = f(a)$ and $m = |w|$. There are three possibilities:

- If $m = |w| = 0$, we introduce an ε-transition from q to $\delta(q, a)$ into N.
- If $m = 1$, we define an transition from q to $\delta(q, a)$, labeled with $f(w)$ into N.
- If $m > 1$, then we insert $m-1$ new states $q_1, \ldots, q_t$ to the N, and connect q to q_1 (with a transition edge marked by w_1), connecting q_i to q_{i+1} with transition edge marked by w_{i+1}, for $i = 1, \ldots, m-2$, and a transition edge marked by w_m from q_{m-1} to $\delta(q, a)$.

We continue this process untill all states and characters from the DFA M had been handled.

Lemma 4.1. *The language $L' = L(N)$ is $f(L)$.*

Proof: Consider a word $x = x_1 \ldots x_t \in L$, and consider its associated path $q_0 = s$, and $q_i = \delta(q_{i-1}, x_i)$, for $i = 1, \ldots, t$. Here $q_t \in A$. Observe, that by construction, there is a path from q_{i-1} to q_i, that can be traversed by the string $f(x_i)$. Furthermore, we have that q_t is in the accepting states of N. This implies that $f(x) = f(x_1)f(x_2)\cdots f(x_t) \in L(N)$, which implies that $f(L) \subseteq L(N)$.

As for the other direction, consider any word $y = y_1, \ldots, y_\ell \in L(N)$, and its associated set of states $q_0, \ldots, q_\ell$. Break this sequence of states into blocks,u where a block starts in a state that is in the original DFA M, and all other states in the block are new and were added to N. It is now straightforward to verify that a block corresponds to a code word, and we can now replace this block by the single character that created it. If the block has a single state, then we insert the symbol that maps to ϵ. The resulting string s, is in the language of the original DFA M. That is $s \in L$, and we conclude that $f(s) = y$. In particular, $y \in f(L)$, implying that $L(N) \subseteq f(L)$.

We conclude that $f(L) = L(N)$. ■

Rubric: 100 points

(a) 10 points.

(b) 20 points. (-5 point for minior error)

(c) 70 points.

- 30 for constructions (standard constructin rubric, scaled).
- 20 for $\subseteq$ and 20 for $\supseteq$

11 (100 PTS.) **Prove Not Regular**

Prove that the following languages in **11.A.** to **11.C.** are not regular by providing a fooling set. You need to prove that it is an infinite fooling and valid fooling set.

11.A. (25 PTS.) $L = \{0^i 1^j 2^k \mid i + j = k + 1\}$.

Solution:

Let F be the language 0^*.

Let x and y be arbitrary strings in F.

Then $x = 0^m$ and $y = 0^n$ for some non-negative integers $m \neq n$.

Let $w = 1^{k+1-m} 2^k$.

Then $xw = 0^m 1^{k+1-m} 2^k \in L$, , because $m + k + 1 - m = k + 1$.

And $yw = 0^n 1^{k+1-m} 2^k \notin L$, because $n + k + 1 - m \neq k + 1$.

Thus, F is a fooling set for L.

Because F is infinite, L cannot be regular

11.B. (25 PTS.) $L = \{0^{n^3} \mid n \geq 0\}$.

Solution:

Let $F = L = \left\{ 0^{n^3} \,\middle|\, n \geq 0 \right\}$.

Let x and y be arbitrary strings in F.

Then $x = 0^{i^3}$ and $y = 0^{j^3}$ for some non-negative integers $i < j$.

Let $w = 0^{3i^2+3i+1}$.

Then $xw = 0^{i^3} 0^{3i^2+3i+1} = 0^{i^3+3i^2+3i+1} = 0^{(i+1)^3} \in L$.

And $yw = 0^{j^3}0^{3i^2+3i+1} = 0^{j^3+3i^2+3i+1} \notin L$, because $i < j$ and the equation cannot be constructed as a cube function. Specifically, the gap between consecutive cubes keeps increasing. So, $j^3 < j^3 + 3i^2 + 3i + 1 < (j+1)^3$

Thus, F is a fooling set for L.

Because F is infinite, L cannot be regular.

11.C. (25 PTS.) $L = \left\{ 0^k w\overline{w} 1^k \;\middle|\; 0 \le k \le 3, w \in \{0,1\}^+ \right\}$, where $\overline{w}$ is the complement bit-wise not operator. For $w = w_1 w_2 \ldots w_m \in \{0,1\}^*$, we define $\overline{w} = \overline{w_1}\,\overline{w_2} \ldots \overline{w_m}$, for $\overline{0} = 1$ and $\overline{1} = 0$.

Solution:

Let F be the language 0^*10.

Let x and y be arbitrary strings in F such that:

$x = 0^i10$ and $y = 0^j10$ for some non-negative integers $j > i > 3$.

Let $s = 1^i$.

Then $xs = 0^i101^i \in L$, because $w = 0^i1$ and $k = 0$.

And $ys = 0^j101^i \notin L$ for any choice of w and k.

Thus, F is a fooling set for L.

Because F is infinite, L cannot be regular

11.D. (25 PTS.) Suppose L is not regular. Show that $L \cup L'$ is not regular for any finite language L'. Give a simple example to show that $L \cup L'$ may be regular if L' is infinite.

Solution:

Let $L'' = L' \backslash L$. Then L'' is regular since it is finite, and all finite languages are regular. Suppose $L \cup L'$ is regular. This implies $L = (L \cup L') \backslash L''$ is regular, since the difference between two regular languages is also regular. This contradicts the fact that L is not regular.

For the example, let $L = \{0^n 1^n \mid n \ge 0\}$ which is not regular and $L' = \{0,1\}^*$ which is infinite. Then $L \cup L' = \{0,1\}^*$ is regular.

Rubric: On a scale of 100 points:

- 75 points for (a,b,c), 25 points for each subquestion (These are not the only correct solutions):
  - 10 point for a proper setup: an infinite fooling set, x, y which are arbitrary pairs in the fooling set, z which is arbitrary string, and proving exactly one of $\{xz, yz\}$ is in L. No further points if this part is incorrect.
  - 15 point for correctly proving z distinguishes x, y.
  - For (b), 5 points for explaining why the expression cannot be written as cube.
- 25 points for (d):
  - 15 points for the proof.
  - 10 points for the example.
- -5 each minor error.

12 (100 PTS.) **Context Free Grammar**

Describe a context free grammar for the following languages in **12.A.** to **12.C.**. Clearly explain how they work and the role of each non-terminal. Unclear grammars will receive little to no credit.

12.A. (20 PTS.) $L_1 = \left\{ 0^i 1^j 2^k 3^\ell 4^t \;\middle|\; i,j,k,\ell,t \geq 0 \text{ and } i+j+k+t = \ell \right\}$.

Solution:

$$
\begin{aligned}
&S \to AE \\
&A \to B \mid 0A3 && // \; L(B) = \left\{ 0^i 1^j 2^k 3^\ell \;\middle|\; \ell = i+j+k \right\} \\
&B \to C \mid 1B3 && // \; L(C) = \left\{ 1^j 2^k 3^\ell \;\middle|\; \ell = j+k \right\} \\
&C \to \varepsilon \mid 2C3 && // \; L(D) = \left\{ 2^k 3^k \;\middle|\; k \geq 0 \right\} \\
&E \to \varepsilon \mid 3E4 && // \; L(E) = \left\{ 3^t 4^t \;\middle|\; t \geq 0 \right\}
\end{aligned}
$$

Here S is the start symbol.

To see why this is correct, observe that for $w = a^i b^j c^k d^\ell e^t$ in the language, we have $i+j+k+t = \ell$, which implies

$$1^i 2^j 3^k 3^\ell 4^t = 1^i 2^j 3^k 3^{i+j+k+t} 4^t = 1^i 2^j 3^k 3^i 3^j 3^k 3^t 4^t = 1^i 2^j 3^k 3^k 3^j 3^i 3^t 4^t = \underbrace{0^i \overbrace{1^j \underbrace{2^k 3^k}_{C} 3^j}^{B} 3^i}_{A} \underbrace{3^t 4^t}_{E}.$$

12.B. (20 PTS.) $L_2 = \{0,1\}^* \setminus \{0^n 1^n \mid n \geq 0\}$, i.e., the complement of the language $\{0^n 1^n \mid n \geq 0\}$.

Solution:

L_2 is the union of the language $L_A = \{0^m 1^n \mid m \neq n, m, n \geq 0\}$ and the language $L_B = (0+1)^* 10 (0+1)^*$. L_A is contained in L_2 by its definition. L_B is contained in L_2 because L_B is the complement of 0^*1^*. 0^*1^* is the union of L_A and $\{0^n 1^n \mid n \geq 0\}$.

On the other hand, $\forall w \in L_2$, w is either in L_A or L_B by the definition of L_2. Since if $w \notin L_A \cup L_B$, then $w \notin L_A$ and $w \notin L_B$. By the definition of L, L_A and L_A, $w \in \{0^n 1^n \mid n \geq 0\}$. This contradicts with the assumption that $w \in L_2$.

$$
\begin{aligned}
&S \to T \mid X && L(S) = \{0,1\}^* \setminus \{0^n 1^n \mid n \geq 0\} \\
\\
&T \to 0T1 \mid A \mid B && L(T) = \{0^m 1^n \mid m \neq n, m, n \geq 0\} \\
&A \to 0 \mid 0A && L(A) = 0^+ \\
&B \to 1 \mid 1B && L(B) = 1^+ \\
\\
&X \to Z10Z && L(X) = (0+1)^* 10 (0+1)^* \\
&Z \to \epsilon \mid 0Z \mid 1Z && L(Z) = (0+1)^*
\end{aligned}
$$

12.C. (20 PTS.) $L_3 = \{0^i 1^j 2^k \mid k = 2(i + j)\}$.

Solution:

$$S \to 0S22 \mid B \qquad\qquad L(S) = \left\{0^i 1^j 2^k \mid k = 2(i + j)\right\}$$
$$B \to 1B22 \mid \varepsilon \qquad\qquad L(B) = \left\{1^j 2^k \mid k = 2j\right\}$$

12.D. (40 PTS.) Prove that your grammar for L_3 in **12.C.** is correct. You need to prove that $L_3 \subseteq L(G)$ and $L(G) \subseteq L_3$ where G is your grammar from part **12.C.**. (See solved problem for an example of how this is done.)

Solution:

We will first prove a separate lemma that we will use in the solution.
Let the language $L' = \left\{1^j 2^k \mid k = 2j\right\}$

Lemma 4.2. *$L' \subseteq L(B)$.*

Proof: Let w be an arbitrary string in L'. By definition, $w = 1^j 2^{2j}$ for some non-negative integer j. Assume that $1^l 2^{2l} \in L(B)$ for every non-negative integer $l < j$. There are two cases to consider.

- If $|w| = 0$, then $1^0 2^0 = \varepsilon$. The rule $B \to \varepsilon$ implies that $B \rightsquigarrow \varepsilon$ and therefore $B \rightsquigarrow^* \varepsilon$.
- Suppose $j > 0$. Then $w = 1^n 2^{2n}$ for some non-negative integer n. Then the first character in w must be 1 and the string must end with 22. The inductive hypothesis implies that $B \rightsquigarrow^* 1^{j-1} 2^{2(j-1)}$. The rule $B \to 1B22$ implies that $B \rightsquigarrow 1B22 \rightsquigarrow^* 1^j 2^{2j}$.
  ■

Lemma 4.3. *$L(B) \subseteq L'$.*

Proof: Let w be an arbitrary string in $L(B)$. Assume that L' contains every string $x \in L(B)$ such that $|x| < |w|$. There are two cases to consider.

- If $|w| = 0$, then $1^0 2^0 = \varepsilon$. The rule $B \to \varepsilon$ implies that $B \rightsquigarrow \varepsilon$ and therefore $B \rightsquigarrow^* \varepsilon$.
- Suppose $|w| > 0$. The inductive hypothesis implies that $B \rightsquigarrow^* 1^{n-1} 2^{2(n-1)}$. The rule $B \to 1B22$ implies that $B \rightsquigarrow 1B22 \rightsquigarrow^* 1^n 2^{2n}$. ■

Together, $L' \subseteq L(B)$ and $L(B) \subseteq L'$ imply that $L' = L(B)$

Lemma 4.4. $L \subseteq L(S)$

Proof: (induction on i) Let w be an arbitrary string in L. By definition, $w = 0^i 1^j 2^{2(i+j)}$ for some non-negative integers i and j. Assume that $0^h 1^j 2^{2(h+j)} \in L(S)$ for all non-negative integers $h < i$. There are two cases to consider.

- If $i = 0$, then $w = 1^j 2^{2j}$. Lemma 1 immediately implies $S \to B \rightsquigarrow^* w$.

- Suppose $i > 0$. Then $w = 0 \bullet 0^{i-1}1^j2^{i+j-2} \bullet 22$. The inductive hypothesis implies that $S \rightsquigarrow^* 0^{i-1}1^j2^{2(i+j)-2} \in L(S)$. It follows that $S \rightsquigarrow 0S22 \rightsquigarrow^* w$.

In both cases, we conclude that $S \rightsquigarrow^* w$. ∎

Proof: (Another proof, this time by induction on $|w|$) Let w be an arbitrary string in L. Assume that $L(S)$ contains every string $x \in L$ such that $|x| < |w|$. There are three cases to consider.

- If $w = \epsilon$, then $S \rightsquigarrow B \rightsquigarrow \epsilon$.
- Suppose $w = 0x$ for some string x. Then $w = 0^i1^j2^{2(i+j)}$ where $i > 0$, so w must end with 22. Thus, we have $w = 0y22$, where $y \in L$. The induction hypothesis implies that $y \in L(S)$. We conclude that $S \rightsquigarrow 0S22 \rightsquigarrow^* w$.
- Suppose $w = 1x$ for some string x. Then $w = 1^j2^{2j}$ for some $j > 0$, and therefore $S \rightsquigarrow B \rightsquigarrow^* w$ by Lemma 1.

In both cases, we conclude that $S \rightsquigarrow^* w$. Note that $|w|$ cannot start with 2, because every string in L that has a 2 has a 0 or 1 before it. ∎

Lemma 4.5. *$L(S) \subseteq L$.*

Proof: Let w be an arbitrary string in $L(S)$. Assume L contains every string $x \in L(S)$ such that $|x| < |w|$. There are two cases to consider

- Suppose $w = 0x22$ for some $x \in L(S)$. The induction hypothesis implies that $x = 0^i1^j2^{(i+j)}$ for some integers i and j. It follows that $w = 0^{i+1}1^j2^{2(i+j)+2}$, and therefore $w \in L$.
- Suppose $w \in L(B)$. Lemma 2 implies that $w = 1^l2^{2l}$ for some integer l. It follows immediately that $w = 0^01^l2^{0+2l} \in L$.

In both cases, we conclude that $w \in L$. ∎

Together, the lemmas imply that $L = L(S)$

<u>*Rubric:*</u> 100 points

(a,b,c) (20 points each)

- 12 for a correct grammar. (These are not the only correct solutions.)
- 8 for a clear explanation of the grammar.
- if the solution is not understandable and no explanation, give 0.

(d) (40 points)

- 20 for $\subseteq$ (standard induction rubric, scaled).
- 20 for $\supseteq$ (standard induction rubric, scaled).