1. \( S_0 \) is the start state. \( S_4 \) is the final state representing the PIN is accepted by the ATM. \( S_1, S_2, S_3 \) represent wrong password typed for 1, 2, 3 times. Once \( S_3 \) is reached, the account is locked so the system is stuck at this state no matter what the user does further.

![State Transition Diagram]

2. There are four states: even 1s and even 0s (e1e0), even 1s and odd 0s (e1o0), odd 1s and even 0s (o1e0), odd 1s and odd 0s (o1o0).

![State Transition Diagram]

3. a) \( \lambda \cup 0 \cup 1 \cup 00 \cup 01 \cup 10 \cup 11 \)

b) \( 001^*0 \)

c) We can interpret it in two ways. Every 1 is followed by exactly two 0s: \( 0^*(100)^* \). Or every 1 is followed by at least two 0s: \( 0^*(1000)^* \). Either interpretation is correct.

d) It is implied that every 1 must be followed by at least a 0. \( 0^*(100)^*(100 \cup 00) \)

e) Since \( 0^*10^*10^* \) represents strings containing two 1s, \( (0^*10^*10^*)^* \) represents strings containing an even number of 1s.
4. We prove this by contradiction. Firstly, assume that the set \( L = \{1^n2|n = 0, 1, 2, \ldots \} \) is regular, i.e., the string \( 1^n2 \) is recognized by \( FSM(S, I, f, s_0, F) \).

Let \( x = 1^n2 \) for some \( n_0 \geq \sqrt{|S|} \), then according to the pumping lemma, we can re-write the \( x \) as \( x = wvu \) with a non-empty \( v \) such that \( wv^k u \) is in the set of \( FSM \) for all \( k \).

Since \( x = 1^n2 \), \( x \) only has ‘1’s in it. Therefore, \( w = 1^a \), \( v = 1^b \), \( u = 1^c \) where \( a, c \) are non-negative integers and \( b \) is a positive integer (since \( v \) should be non-empty). Then, for any \( i = 0, 1, \ldots \), \( wv^k u = 1^a(1^b)^k(1^c) = 1^{a+c+bk} \) is in \( L \). This implies that \( a + c + bi \) is a perfect square for all \( i = 0, 1, \ldots \), which is impossible since \( a + c + bi \) has a linear growth while perfect squares have a quadratic growth.

Therefore, our assumption that the set is regular is wrong, which means the set is not regular.

Extra credit problems

1. (a)

**Base Case**: easily verified, thus omitted.

**Assumption**:
Assume \( \varphi_{2i}(a) = H_{2i}a, \varphi_{2i+1}(a) = H_{2i+1} \bar{c} \).

**Step**:
\[
\begin{align*}
\varphi_{2i+2}(a) &= \varphi^2(a)\varphi^2(a) = H_{2i}aH_2a = H_{2i+2}a \\
\varphi_{2i+3}(a) &= \varphi^2(a)\varphi^3(a) = H_{2i}aH_3\bar{c} = H_{2i+3}\bar{c}
\end{align*}
\]

In conclusion, \( \varphi_{2i}(a) = H_{2i}a, \varphi_{2i+1}(a) = H_{2i+1}\bar{c} \) for \( i \geq 1 \).

1. (b)

When \( N \) is even:
\[
\lim_{N \rightarrow \infty} \varphi^N(a) = \lim_{N \rightarrow \infty} H_Na = \lim_{N \rightarrow \infty} H_N = H
\]

When \( N \) is odd:
\[
\lim_{N \rightarrow \infty} \varphi^N(a) = \lim_{N \rightarrow \infty} H_N\bar{c} = \lim_{N \rightarrow \infty} H_N = H
\]

2.

The proof follows by the induction below.
\[
\begin{align*}
H_{2N+1} &= (aCbAcB)^{2^{2N+1}-2}a, \text{ for } N \geq 0 \\
H_{2N} &= (aCbAcB)^{2^{2N}-4}aCb, \text{ for } N \geq 1
\end{align*}
\]
3. 

\[ \varphi(\sigma(a)) = \varphi(b) = cb = \sigma^{-1}(a\tilde{c}) = \sigma^{-1}(\varphi(a)) \]

\[ \varphi(\sigma^{-1}(a)) = \varphi(c) = b\tilde{a} = \sigma(a\tilde{c}) = \sigma(\varphi(a)) \]

Similarly, it can be shown that the equations hold for every letter in \( \Sigma \). Then, because of the linearity of the functions, the equations hold for all strings in \( \Sigma \).

4.(a) 

**Base case:** easily verified, thus omitted.

**Assumption:**

\[ \varphi(H_{2i})a = H_{2i+1}, \varphi(H_{2i+1})b = H_{2i+2} \]

**Step:**

\[ \varphi(H_{2i+2})a = \varphi(H_{2i+1}c\sigma(H_{2i+1}))a \]

\[ = \varphi(H_{2i+1})ba\varphi(\sigma(H_{2i+1}))a \]

\[ = \varphi(H_{2i+1})ba\sigma^{-1}(\varphi(H_{2i+1})b) \]

\[ = H_{2i+2}a\sigma^{-1}(H_{2i+2}) \]

\[ = H_{2i+1} \]

Similarly,

\[ \varphi(H_{2i+3})a = \varphi(H_{2i+2}a\sigma^{-1}(H_{2i+2}))b \]

\[ = \varphi(H_{2i+2})a\tilde{c}\varphi(\sigma^{-1}(H_{2i+2}))b \]

\[ = \varphi(H_{2i+2})a\tilde{c}\sigma(\varphi(H_{2i+2})a) \]

\[ = H_{2i+3}c\sigma(H_{2i+3}) \]

\[ = H_{2i+4} \]

In conclusion, the following is true for all \( i > 0 \).

\[ \varphi(H_{2i})a = H_{2i+1}, \varphi(H_{2i+1})b = H_{2i+2} \]

4.(b) 

\( \varphi(H_{2i}) \) is a prefix of length \( 2^{2i+1} - 2 \) of \( H_{2i+1} \), therefore, \( H_{2i} \) is mapped to a longer prefix of \( H \). Also, \( \varphi(H_{2i+1}) \) is a prefix of length \( 2^{2N} - 4 \) of \( H_{2i+2} \), and \( H_{2i+1} \) is also mapped to a longer prefix of \( H \). Both combined, we have the conclusion that \( \phi(H) \) has to be equal to \( H \).

5.
Table 1: Verification for 10 state transitions

<table>
<thead>
<tr>
<th>Current state $x$</th>
<th>$\varphi(x)$</th>
<th>State for 0</th>
<th>State for 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a$</td>
<td>$ac\bar{c}$</td>
<td>$a$</td>
<td>$\bar{c}$</td>
</tr>
<tr>
<td>$b$</td>
<td>$c\bar{b}$</td>
<td>$c$</td>
<td>$\bar{b}$</td>
</tr>
<tr>
<td>$c$</td>
<td>$b\bar{a}$</td>
<td>$b$</td>
<td>$\bar{a}$</td>
</tr>
<tr>
<td>$\bar{b}$</td>
<td>$cb$</td>
<td>$c$</td>
<td>$b$</td>
</tr>
<tr>
<td>$\bar{c}$</td>
<td>$ba$</td>
<td>$b$</td>
<td>$a$</td>
</tr>
</tbody>
</table>