1.21

(a) Here are the successive phases of state elimination:

\[ \rightarrow O \xrightarrow{\varepsilon} (1) \overset{a}{\xleftarrow{b}} (2) \xrightarrow{\varepsilon} O \]

Eliminate state 2:

\[ (1) \overset{b}{\rightarrow} (2) \overset{\varepsilon}{\rightarrow} (1) \Rightarrow (1) \overset{\text{au}ba^*b}{\rightarrow} (1) \]

\[ (1) \text{ ba}^* \Rightarrow (1) \]

Regular expression:

\[(\text{au}ba^*b)^* \text{ ba}^*\]

The language is the set of strings with an odd number of b's.

(If I were just to guess this without the automaton, I would write

\[a^* b (a^* b a^* b a^*)^*\]
1. 21(b)

\[
\rightarrow 0 \overset{\varepsilon}{\rightarrow} 1 \overset{a, b}{\rightarrow} 2 \overset{b}{\leftrightarrow} 3 \overset{a}{\rightarrow} 1 \overset{\varepsilon}{\rightarrow} 0
\]

Eliminate 2:
\[
\begin{align*}
(1) & \quad a \cup b \rightarrow 2 \overset{b}{\rightarrow} 3 \quad \Rightarrow \quad (a \cup b) a^* b \\
(3) & \quad b \rightarrow 2 \overset{a}{\rightarrow} 3 \quad \Rightarrow \quad 3 \overset{b a^* b}{\rightarrow} 3
\end{align*}
\]

\[
\rightarrow 0 \overset{\varepsilon}{\rightarrow} 1 \overset{(a \cup b) a^* b}{\leftarrow} 3 \overset{b a^* b}{\rightarrow} 3 \overset{\varepsilon}{\rightarrow} 0
\]

Eliminate 3:
\[
\begin{align*}
(1) & \quad (a \cup b) a^* b \rightarrow 3 \overset{b a^* b}{\rightarrow} 0 \quad \Rightarrow \quad 1 \overset{(a \cup b) a^* b (b a^* b)^*}{\rightarrow} 0 \\
(1) & \quad (a \cup b) a^* b \rightarrow 3 \overset{b a^* b}{\rightarrow} 1
\end{align*}
\]

\[
\rightarrow 0 \overset{\varepsilon}{\rightarrow} 1 \overset{\varepsilon \cup X}{\rightarrow} 1 \text{ where } X = (a \cup b) a^* b (b a^* b)^*
\]

Regular expression:
\[
((a \cup b) a^* b (b a^* b)^*)^* (\varepsilon \cup X) = (a \cup b) a^* b (b a^* b)^*
\]

Not on oath!
(b) To simplify the computation, observe that we can just eliminate the state at lower left, along with its incoming arrows. We then eliminate state $y$ by

$$\text{aulpu} → [\text{aulpu}^*]$$

and the r.e.

$$\# [\text{aulpu}^*]^{**}$$
1.29(b)
Suppose $A_2 = \{www: w \in \{a,b\}^*\}$ is regular.
Let $N$ be the number given in the pumping lemma. Consider $W = a^nba^nb a^nba^n$.

$W \in L$, and by the pumping lemma

$w = xyz$ where $|xy| \leq N$

$|y| > 0$

and $xz \in L$.

We now have $x = a^k$, $y = a^r$, $z = a^{n-k}ba^nb a^n$

with $r > 0$.

$xz = a^{n-r}ba^nb a^n$

which is not of the form $vvv$ for any string $v$, so $xz \notin L$, a contradiction.

Thus $A_2$ is not regular.

1.46(c) $L = \{w: w \in \{0,1\}^* \text{ is not a palindrome}\}$

Suppose $L$ is regular. Then its complement, the set $P$ of palindromes, would also be regular, and so would $P \cap 0^*10^* = \{0^n10^n: n \geq 0\}$.

But we prove this is nonregular using the pumping lemma, (the argument is exactly the same as Ex. 1.73) so $L$ is not regular.
1.48 A string is in \( D \) if the number of 'switches' from 0 to 1 is equal to the number of switches from 1 to 0. But the switches must alternate, so in fact \( D \) consists of the strings over \( \{0,1\} \) that start and end with the same symbol: \( 0(01)^*0 \cup 1(01)^*1 \).

1.53: \( ADD \& 4^*+0=1^* \)

Suppose \( ADD \) is regular. Then so is its inverse, with the regular language \( (4^*0=1^*) \), which is \( \{1^n4^*0=1^n : n \geq 0\} \). But we can prove this is nonregular using precisely the same argument as Example 1.73.

grep expressions. (with number of hits on my imac dictionary)

(a) At least 2 consecutive 2's:
   \[ *2.* \] (128 words)

(b) At least 2 g's and 1 m:
   \[
   \begin{align*}
   &(*m.*g.*g.*)| (.*g.*m.*g.*) | (.*g.*g.*m.*)
   \end{align*}
   \] (426 words)

(c) At least 3 i's and no other vowels (here I count y as a vowel):
   \[
   \begin{align*}
   &[^aeouy]*i [*aeouy]^+i [*aeouy]^+i [*aeouy]^* \end{align*}
   \] (225 words)
(1) Contains none of the vowels a, e, i, o, u

\[ [\text{aeiou}]^* \quad \text{(may be better to try \([\text{NaAcEiIoUnU}]^*\) with uppercase letters)} \]

(2) contains at least six letters that are either s or t.

\[ \cdot * (s/t), * (s/t), * (s/t), * (s/t), * (s/t), * (s/t), * \]

(227 hits)