Exercise 2.2.6 Solution

a. Please see the web for explanation

http://www-db.stanford.edu/~ullman/ralc.html

b. Let's take a little example

\[ 10011 = 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \]

\[ = 25 \text{ (this is divisible by 5 and should be accepted by the automaton)} \]

Also observe the property: \((a+b) \mod 5 = a \mod 5 + b \mod 5\).

Therefore, for our example above

\[ (1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0) \mod 5 \]

\[ = 1 \times 2^5 \mod 5 + 0 \times 2^4 \mod 5 + 0 \times 2^3 \mod 5 + 1 \times 2^2 \mod 5 + 1 \times 2^1 \mod 5 + 1 \times 2^0 \mod 5 \]

\[ = 1 \mod 5 + 0 \mod 5 + 0 \mod 5 + 1 \mod 5 + 1 \mod 5 + 1 \mod 5 \]

\[ = 1 + 0 + 0 + 1 + 1 + 1 \]

\[ = 5 \text{ (which is also still divisible by 5)} \]

Assume now our string is abcdefgh..., where each alphabet stand for 0 or 1.

To find out whether abcdefgh..., is divisible by 5, what we do calculate

\[ (a \times 2^6 + b \times 2^5 + c \times 2^4 + d \times 2^3 + e \times 2^2 + f \times 2^1 + g \times 2^0 + h \times 2^{-1} + \ldots) \mod 5 \]

\[ = a \times 2^6 \mod 5 + b \times 2^5 \mod 5 + c \times 2^4 \mod 5 + d \times 2^3 \mod 5 + e \times 2^2 \mod 5 + f \times 2^1 \mod 5 + g \times 2^0 \mod 5 + h \times 2^{-1} \mod 5 + \ldots \]
\[ e \times 2^4 \mod 5 + f \times 2^5 \mod 5 + g \times 2^6 \mod 5 + h \times 2^7 \mod 5 + \ldots \]

Now notice that

\[
\begin{align*}
2^0 \mod 5 &= 1 \\
2^1 \mod 5 &= 2 \\
2^2 \mod 5 &= 4 \\
2^3 \mod 5 &= 3 \\
2^4 \mod 5 &= 1 & \text{(the cycle goes back to 1)} \\
2^5 \mod 5 &= 2 \\
2^6 \mod 5 &= 4
\end{align*}
\]

will just repeat as before.

So, our equation above becomes

\[
= a \times 1 + b \times 2 + c \times 3 + \\
\text{ } e \times 1 + f \times 2 + g \times 3 + \ldots
\]

(Where recall that a, b, c, ... just stands for either 0 or 1)

Let's consider our example again: 10011

When we first see input 1, we calculate \( 1 \times 1 = 1 \). So our sum so far is 1.
The next input is 0, and since \( 0 \times 2 = 0 \), our current sum is still 1. But remember that we already use the 'times 2' level, so the next input will use the 'times 4' level. The next input is still 0, and since \( 0 \times 4 = 0 \), our sum is still 1.
The next input is 1, and now \( 1 \times 3 = 3 \), and thus our current sum is \( 1 + 3 = 4 \). It's very important that we now are...
using the x3 level. That's the reason why we keep tracking our current level even though the input bit is 0. The next input is 1, and we back at x1 level, so 1 \times 1 = 1 and our current sum is 4 + 1 = 5, which should be accepted by the automata.

Note: $q^{\text{super}}_{\text{sub}}$ indicates that our current sum is equal to sub.

This means that if the next input is $\text{a (either 0 or 1)}$ and the next level is $\text{e}$, we are going to compute $(\text{a} \times \text{e}) \mod 5 + \text{current sum}$ to get our current sum.