Exercise 4.2.6

Show that the regular languages are closed under the operations below. For each, we’ll start with \(L\) and apply operations under which regular languages are closed (homomorphisms, intersection, set difference) to get the desired language.

a) \(\text{min}(L) = \{w \mid w \text{ is in } L, \text{but no proper prefix of } w \text{ is in } L\}\)

Describe the strings which are ineligible for \(\text{min}(L)\) and exclude them using set difference. The ineligible strings are \(L\Sigma^+\), since \(w \in L\Sigma^+\) means that \(w = xy\) where \(x \in L\). That is, \(w\) has a proper prefix \(x\) which is in \(L\).

\[
\text{min}(L) = L - L\Sigma^+
\]

Comments: many students submitted the (correct) answer \(L - (L \cap L\Sigma^+)\). This probably looks more natural, since \(L \cap L\Sigma^+\) is a subset of \(L\), and \(L\Sigma^+\) is not (in general). However, recall the definition of set difference: \(L - M = L \cap \overline{M}\). Think of this not as removing \(M\) from \(L\), but excluding \(M\) from \(L\), since \(M\) need not be a subset of \(L\).

Also, many students submitted a solution using homomorphisms, which were not necessary. It should be a red flag if you define a pair of homomorphisms that just add and remove hats without otherwise altering the string. This means that your set intersection could be done in the original alphabet \(\Sigma\).

b) \(\text{max}(L) = \{w \mid w \text{ is in } L \text{ and for no } x \text{ other than } \epsilon \text{ is } wx \text{ in } L\}\)

Again, describe strings which are ineligible for \(\text{max}(L)\), then remove them with set difference. Use homomorphisms to describe all the strings that are prefixes of strings in \(L\). Here is the idea: use an inverse homomorphism \(h^{-1}\) on \(L\) to mark arbitrary symbols with hats, apply set intersection to force the hats to the end, then apply a homomorphism \(g\) to delete hatted symbols. For simplicity, assume the alphabet is \(\{a, b\}\) (the approach works for any alphabet).

\[
\begin{align*}
\text{h}(a) &= a & \text{g}(a) &= a \\
\text{h}(\hat{a}) &= a & \text{g}(\hat{a}) &= \epsilon \\
\text{h}(b) &= b & \text{g}(b) &= b \\
\text{h}(\hat{b}) &= b & \text{g}(\hat{b}) &= \epsilon
\end{align*}
\]

Then \((a + b)^*(\hat{a} + \hat{b})^+\) is the expression describing the strings with hats at the end. Our final expression is:

\[
\text{max}(L) = L - g(h^{-1}(L) \cap (a + b)^*(\hat{a} + \hat{b})^+))
\]
c) $init(L) = \{w \mid \text{for some } x, wx \text{ is in } L\}$

Note that this is almost exactly the set of strings we declared ineligible for $max(L)$, except that in $init(L)$, $x$ may be $\epsilon$. Use the same homomorphisms $h$ and $g$, but modify $E$ to allow omission of the hatted portion:

$$init(L) = g(h^{-1}(L) \cap (a + b)^* (\hat{a} + \hat{b})^*)$$