4. Let $K = \{01, 01001, 010010001, \ldots \}$. We assume that all strings in $K$ start with 01. What we want to find is what looks like a regular expression for $\bar{K}$, the complement of $K$.

Note that we can’t actually write down a real regular expression for $\bar{K}$ since this language is not regular (and neither is $K$ itself). This is where the need for $L$ comes in. It will be the only non-regular part of our final expression.

The easy cases: A string $w$ can’t be in $K$ (and thus in $\bar{K}$) if $w$

1. begins with 1 ($\Rightarrow 1(0|1)^*$),
2. ends with 0 ($\Rightarrow (0|1)^*0$),
3. begins with 00 ($\Rightarrow 00(0|1)^*$), or
4. is $\varepsilon$.

The more interesting cases (where we make use of $L$): The key observation is that if we ever see a block of the form $0^i10^j$ in $w$, where $j \neq i + 1$, we know $w$ is not in $K$. This can be expressed using $L$ as $0^*L|L000^*$. A subtle detail is that we need to make sure no more 0s are tacked onto the wrong end of such a block, potentially invalidating our pattern. One possible solution is to write

$$
(0|1)^*(L1|1L00)(0|1)^* | L00(0|1)^*.
$$

So the full expression for $\bar{K}$ is

$$
(0|1)^*(L1|1L00)(0|1)^* | L00(0|1)^* | 1(0|1)^* | (0|1)^*0 | 00(0|1)^* | \varepsilon.
$$