d) If L is the language $(\mathbf{1}+\mathbf{02})^*$ what is $h(L)$?

* e) If L is the language $(ab+baa)^*bab$, what is $h^{-1}(L)$?

! f) If L is the language consisting of the single string $ababb$ what is $h^{-1}(L)$?

*! **Exercise 4.2.2:** If L is a language, and a is a symbol, then L/a, the *quotient* of L and a, is the set of strings w such that wa is in L. For example, if $L = \{a, aab, baa\}$, then $L/a = \{\epsilon, ba\}$. Prove that if L is regular, so is L/a. *Hint*: Start with a DFA for L and consider the set of accepting states.

! **Exercise 4.2.3:** Lf L is a language, and a is a symbol, then $a\backslash L$ is the set of strings w such that aw is in L. For example, if $L = \{a, aab, baa\}$, then $a\backslash L = \{\epsilon, ab\}$. Prove that if L is regular, so is $a\backslash L$. *Hint*: Remember that the regular languages are closed under reversal and under the quotient operation of Exercise 4.2.2.

! **Exercise 4.2.4:** Which of the following identities are true?

a) $(L/a)a = L$ (the left side represents the concatenation of the languages L/a and $\{a\}$).

b) $a(a\backslash L) = L$ (again, concatenation with $\{a\}$, this time on the left, is intended).

c) $(La)/a = L$.

d) $a\backslash(aL) = L$.

Exercise 4.2.5: The operation of Exercise 4.2.3 is sometimes viewed as a "derivative," and $a\backslash L$ is written $\frac{dL}{da}$. These derivatives apply to regular expressions in a manner similar to the way ordinary derivatives apply to arithmetic expressions. Thus, if R is a regular expression, we shall use $\frac{dR}{da}$ to mean the same as $\frac{dL}{da}$, if $L = L(R)$.

a) Show that $\frac{d(R+S)}{da} = \frac{dR}{da} + \frac{dS}{da}$.

*! b) Give the rule for the "derivative" of RS. *Hint*: You need to consider two cases: if $L(R)$ does or does not contain ϵ. This rule is not quite the same as the "product rule" for ordinary derivatives, but is similar.

! c) Give the rule for the "derivative" of a closure, i.e., $\frac{d(R^*)}{da}$.

d) Use the rules (a), (b) and (c) to find the derivatives of regular expression $\mathbf{1}(\mathbf{01}+\mathbf{10})^*\mathbf{00}$ with respect to 0 and 1.

* e) Characterize those languages for which $\dfrac{dL}{d0} = \epsilon$.

*! f) Characterize those languages for which $\dfrac{dL}{d1} = \{011\}$.

! **Exercise 4.2.6:** Show that the regular languages are closed under the following operations:

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

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

c) $init(L) = \{w \mid$ for some x, wx is in $L\}$.

Hint: Like Exercise 4.2.2, it is easiest to start with a DFA for L and perform a construction to get the desired language.

! **Exercise 4.2.7:** If $w = a_1a_2\cdots a_n$ and $x = b_1b_2\cdots b_m$ are strings of the same length, define $alt(w,x)$ to be the string in which the symbols of w and x alternate, starting with w, that is, $a_1b_1a_2b_2\cdots a_nb_n$. If L and M are languages, define $alt(L,M)$ to be the set of strings of the form $alt(w,x)$, where w is any string in L and x is any string in M of the same length. Prove that if L and M are regular, so is $alt(L,M)$.

*!! **Exercise 4.2.8:** Let L be a language. Define $half(L)$ to be the set of first halves of strings in L, that is, $\{w \mid$ for some x such that $|x| = |w|$, we have wx in $L\}$. For example, if $L = \{\epsilon, 0010, 011, 010110\}$ then $half(L) = \{\epsilon, 00, 010\}$. Notice that odd-length strings do not contribute to $half(L)$. Prove that if L is a regular language, so is $half(L)$.

!! **Exercise 4.2.9:** We can generalize Exercise 4.2.8 to a number of functions that determine how much of the string we take. If f is a function of integers, define $f(L)$ to be $\{w \mid$ for some x, with $|x| = f(|w|)$, we have wx in $L\}$. For instance, the operation $half$ corresponds to f being the identity function $f(n) = n$, since $half(L)$ is defined by having $|x| = |w|$. Show that if L is a regular language, then so is $f(L)$, if f is one of the following functions:

a) $f(n) = 2n$ (i.e., take the first thirds of strings).

b) $f(n) = n^2$ (i.e., the amount we take has length equal to the square root of what we do not take.

c) $f(n) = 2^n$ (i.e., what we take has length equal to the logarithm of what we leave).

!! **Exercise 4.2.10:** Suppose that L is any language, not necessarily regular, whose alphabet is $\{0\}$; i.e., the strings of L consist of 0's only. Prove that L^* is regular. *Hint*: At first, this theorem sounds preposterous. However, an example will help you see why it is true. Consider the language $L = \{0^i \mid i$ is prime$\}$, which we know is not regular by Example 4.3. Strings 00 and 000 are in L, since 2 and 3 are both primes. Thus, if $j \geq 2$, we can show 0^j is in L^*. If j is even, use $j/2$ copies of 00, and if j is odd, use one copy of 000 and $(j-3)/2$ copies of 00. Thus, $L^* = \mathbf{000}^*$.

!! **Exercise 4.2.11:** Show that the regular languages are closed under the following operation: $cycle(L) = \{w \mid$ we can write w as $w = xy$, such that yx is in $L\}$. For example, if $L = \{01, 011\}$, then $cycle(L) = \{01, 10, 011, 110, 101\}$. *Hint*: Start with a DFA for L and construct an ϵ-NFA for $cycle(L)$.

!! **Exercise 4.2.12:** Let $w_1 = a_0a_0a_1$, and $w_i = w_{i-1}w_{i-1}a_i$ for all $i > 1$. For instance, $w_3 = a_0a_0a_1a_0a_0a_1a_2a_0a_0a_1a_0a_0a_1a_2a_3$. The shortest regular expression for the language $L_n = \{w_n\}$, i.e., the language consisting of the one string w_n, is the string w_n itself, and the length of this expression is $2^{n+1} - 1$. However, if we allow the intersection operator, we can write an expression for L_n whose length is $O(n^2)$. Find such an expression. *Hint*: Find n languages, each with regular expressions of length $O(n)$, whose intersection is L_n.

! **Exercise 4.2.13:** We can use closure properties to help prove certain languages are not regular. Start with the fact that the language

$$L_{0n1n} = \{0^n 1^n \mid n \geq 0\}$$

is not a regular set. Prove the following languages not to be regular by transforming them, using operations known to preserve regularity, to L_{0n1n}:

* a) $\{0^i 1^j \mid i \neq j\}$.

b) $\{0^n 1^m 2^{n-m} \mid n \geq m \geq 0\}$.

Exercise 4.2.14: In Section 4.2.2, we described the "product construction" which took two DFA's and constructed another DFA whose language is the intersection of the first two. In a similar way we do a "union construction."

a) Modify the construction to find the union of two DFAs.

! b) Modify the construction to find the union of two NFAs.

* c) Modify the construction to find the Symmetric difference of two DFAs.

d) Modify the construction to find the union of two ε-NFAs.

Exercise 4.2.15: In the proof of Theorem 4.8, we claimed that it could be proved by induction on the length of w that

$$\hat{\delta}\big((q_L, q_M), w\big) = \big(\hat{\delta}_L(q_L, w), \hat{\delta}_M(q_M, w)\big)$$

Give this inductive proof.

Exercise 4.2.16: Complete the proof of Theorem 4.14 by considering the cases where expression E is a concatenation of two subexpressions and where E is the closure of an expression.

Exercise 4.2.17: In Theorem 4.16, we omitted a proof by induction on the length of w that $\hat{\gamma}(q_0, w) = \hat{\delta}(q_0, h(w))$. Prove this statement.