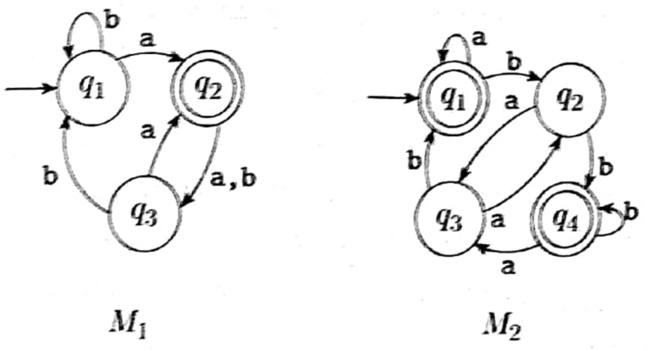
EXERCISES

[A]1.1 The following are the state diagrams of two DFAs, M_1 and M_2. Answer the following questions about each of these machines.



a. What is the start state?

b. What is the set of accept states?

c. What sequence of states does the machine go through on input aabb?

d. Does the machine accept the string aabb?

e. Does the machine accept the string ε?

[A]1.2 Give the formal description of the machines M_1 and M_2 pictured in Exercise 1.1.

1.3 The formal description of a DFA M is $(\{q_1, q_2, q_3, q_4, q_5\}, \{\text{u}, \text{d}\}, \delta, q_3, \{q_3\})$, where δ is given by the following table. Give the state diagram of this machine.

	u	d
q_1	q_1	q_2
q_2	q_1	q_3
q_3	q_2	q_4
q_4	q_3	q_5
q_5	q_4	q_5

1.4 Each of the following languages is the intersection of two simpler languages. In each part, construct DFAs for the simpler languages, then combine them using the construction discussed in footnote 3 (page 46) to give the state diagram of a DFA for the language given. In all parts, $\Sigma = \{\text{a}, \text{b}\}$.

a. $\{w \mid w$ has at least three a's and at least two b's$\}$

[A]b. $\{w \mid w$ has exactly two a's and at least two b's$\}$

c. $\{w \mid w$ has an even number of a's and one or two b's$\}$

[A]d. $\{w \mid w$ has an even number of a's and each a is followed by at least one b$\}$

e. $\{w \mid w$ starts with an a and has at most one b$\}$

f. $\{w \mid w$ has an odd number of a's and ends with a b$\}$

g. $\{w \mid w$ has even length and an odd number of a's$\}$

1.5 Each of the following languages is the complement of a simpler language. In each part, construct a DFA for the simpler language, then use it to give the state diagram of a DFA for the language given. In all parts, $\Sigma = \{a, b\}$.

- [A]**a.** $\{w \mid w$ does not contain the substring $ab\}$
- [A]**b.** $\{w \mid w$ does not contain the substring $baba\}$
- **c.** $\{w \mid w$ contains neither the substrings ab nor $ba\}$
- **d.** $\{w \mid w$ is any string not in $a^*b^*\}$
- **e.** $\{w \mid w$ is any string not in $(ab^+)^*\}$
- **f.** $\{w \mid w$ is any string not in $a^* \cup b^*\}$
- **g.** $\{w \mid w$ is any string that doesn't contain exactly two a's$\}$
- **h.** $\{w \mid w$ is any string except a and $b\}$

1.6 Give state diagrams of DFAs recognizing the following languages. In all parts, the alphabet is $\{0,1\}$.

- **a.** $\{w \mid w$ begins with a 1 and ends with a $0\}$
- **b.** $\{w \mid w$ contains at least three 1s$\}$
- **c.** $\{w \mid w$ contains the substring 0101 (i.e., $w = x0101y$ for some x and y)$\}$
- **d.** $\{w \mid w$ has length at least 3 and its third symbol is a $0\}$
- **e.** $\{w \mid w$ starts with 0 and has odd length, or starts with 1 and has even length$\}$
- **f.** $\{w \mid w$ doesn't contain the substring $110\}$
- **g.** $\{w \mid$ the length of w is at most $5\}$
- **h.** $\{w \mid w$ is any string except 11 and $111\}$
- **i.** $\{w \mid$ every odd position of w is a $1\}$
- **j.** $\{w \mid w$ contains at least two 0s and at most one $1\}$
- **k.** $\{\varepsilon, 0\}$
- **l.** $\{w \mid w$ contains an even number of 0s, or contains exactly two 1s$\}$
- **m.** The empty set
- **n.** All strings except the empty string

1.7 Give state diagrams of NFAs with the specified number of states recognizing each of the following languages. In all parts, the alphabet is $\{0,1\}$.

- [A]**a.** The language $\{w \mid w$ ends with $00\}$ with three states
- **b.** The language of Exercise 1.6c with five states
- **c.** The language of Exercise 1.6l with six states
- **d.** The language $\{0\}$ with two states
- **e.** The language $0^*1^*0^+$ with three states
- [A]**f.** The language $1^*(001^+)^*$ with three states
- **g.** The language $\{\varepsilon\}$ with one state
- **h.** The language 0^* with one state

1.8 Use the construction in the proof of Theorem 1.45 to give the state diagrams of NFAs recognizing the union of the languages described in

- **a.** Exercises 1.6a and 1.6b.
- **b.** Exercises 1.6c and 1.6f.

1.9 Use the construction in the proof of Theorem 1.47 to give the state diagrams of NFAs recognizing the concatenation of the languages described in

a. Exercises 1.6g and 1.6i.

b. Exercises 1.6b and 1.6m.

1.10 Use the construction in the proof of Theorem 1.49 to give the state diagrams of NFAs recognizing the star of the languages described in

a. Exercise 1.6b.

b. Exercise 1.6j.

c. Exercise 1.6m.

A1.11 Prove that every NFA can be converted to an equivalent one that has a single accept state.

1.12 Let $D = \{w|\ w$ contains an even number of a's and an odd number of b's and does not contain the substring ab$\}$. Give a DFA with five states that recognizes D and a regular expression that generates D. (Suggestion: Describe D more simply.)

1.13 Let F be the language of all strings over $\{0,1\}$ that do not contain a pair of 1s that are separated by an odd number of symbols. Give the state diagram of a DFA with five states that recognizes F. (You may find it helpful first to find a 4-state NFA for the complement of F.)

1.14 a. Show that if M is a DFA that recognizes language B, swapping the accept and nonaccept states in M yields a new DFA recognizing the complement of B. Conclude that the class of regular languages is closed under complement.

b. Show by giving an example that if M is an NFA that recognizes language C, swapping the accept and nonaccept states in M doesn't necessarily yield a new NFA that recognizes the complement of C. Is the class of languages recognized by NFAs closed under complement? Explain your answer.

1.15 Give a counterexample to show that the following construction fails to prove Theorem 1.49, the closure of the class of regular languages under the star operation.[7] Let $N_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ recognize A_1. Construct $N = (Q_1, \Sigma, \delta, q_1, F)$ as follows. N is supposed to recognize A_1^*.

a. The states of N are the states of N_1.

b. The start state of N is the same as the start state of N_1.

c. $F = \{q_1\} \cup F_1$.
The accept states F are the old accept states plus its start state.

d. Define δ so that for any $q \in Q_1$ and any $a \in \Sigma_\varepsilon$,

$$\delta(q,a) = \begin{cases} \delta_1(q,a) & q \notin F_1 \text{ or } a \neq \varepsilon \\ \delta_1(q,a) \cup \{q_1\} & q \in F_1 \text{ and } a = \varepsilon. \end{cases}$$

(Suggestion: Show this construction graphically, as in Figure 1.50.)

[7] In other words, you must present a finite automaton, N_1, for which the constructed automaton N does not recognize the star of N_1's language.

[A]a. 0001^*

[A]b. 0^*1^*

c. $001 \cup 0^*1^*$

[A]d. $0^*1^+0^+1^* \cup 10^*1$

e. $(01)^*$

f. ε

g. $1^*01^*01^*$

h. $10(11^*0)^*0$

i. 1011

j. Σ^*

1.51 Prove that the following languages are not regular. You may use the pumping lemma and the closure of the class of regular languages under union, intersection, and complement.

a. $\{0^n1^m0^n \mid m, n \geq 0\}$

[A]b. $\{0^m1^n \mid m \neq n\}$

c. $\{w \mid w \in \{0,1\}^*$ is not a palindrome$\}$[8]

*d. $\{wtw \mid w, t \in \{0,1\}^+\}$

1.52 Let $\Sigma = \{1, \#\}$ and let

$$Y = \{w \mid w = x_1\#x_2\#\cdots\#x_k \text{ for } k \geq 0, \text{ each } x_i \in 1^*, \text{ and } x_i \neq x_j \text{ for } i \neq j\}.$$

Prove that Y is not regular.

1.53 Let $\Sigma = \{0,1\}$ and let

$$D = \{w \mid w \text{ contains an equal number of occurrences of the substrings 01 and 10}\}.$$

Thus $101 \in D$ because 101 contains a single 01 and a single 10, but $1010 \notin D$ because 1010 contains two 10s and one 01. Show that D is a regular language.

1.54 Let $\Sigma = \{a, b\}$. For each $k \geq 1$, let C_k be the language consisting of all strings that contain an a exactly k places from the right-hand end. Thus $C_k = \Sigma^* a \Sigma^{k-1}$. Describe an NFA with $k + 1$ states that recognizes C_k in terms of both a state diagram and a formal description.

1.55 Consider the languages C_k defined in Problem 1.54. Prove that for each k, no DFA can recognize C_k with fewer than 2^k states.

1.56 Let $\Sigma = \{a, b\}$. For each $k \geq 1$, let D_k be the language consisting of all strings that have at least one a among the last k symbols. Thus $D_k = \Sigma^* a (\Sigma \cup \varepsilon)^{k-1}$. Describe a DFA with at most $k+1$ states that recognizes D_k in terms of both a state diagram and a formal description.

*1.57

a. Let A be an infinite regular language. Prove that A can be split into two infinite disjoint regular subsets.

b. Let B and D be two languages. Write $B \Subset D$ if $B \subseteq D$ and D contains infinitely many strings that are not in B. Show that if B and D are two regular languages where $B \Subset D$, then we can find a regular language C where $B \Subset C \Subset D$.

[8] A *palindrome* is a string that reads the same forward and backward.

1.58 Let N be an NFA with k states that recognizes some language A.

a. Show that if A is nonempty, A contains some string of length at most k.

b. Show, by giving an example, that part (a) is not necessarily true if you replace both A's by $\overline{A}$.

c. Show that if $\overline{A}$ is nonempty, $\overline{A}$ contains some string of length at most 2^k.

d. Show that the bound given in part (c) is nearly tight; that is, for each k, demonstrate an NFA recognizing a language A_k where $\overline{A_k}$ is nonempty and where $\overline{A_k}$'s shortest member strings are of length exponential in k. Come as close to the bound in (c) as you can.

***1.59** Prove that for each $n > 0$, a language B_n exists where

a. B_n is recognizable by an NFA that has n states, and

b. if $B_n = A_1 \cup \cdots \cup A_k$, for regular languages A_i, then at least one of the A_i requires a DFA with exponentially many states.

1.60 A ***homomorphism*** is a function $f\colon \Sigma \longrightarrow \Gamma^*$ from one alphabet to strings over another alphabet. We can extend f to operate on strings by defining $f(w) = f(w_1)f(w_2)\cdots f(w_n)$, where $w = w_1 w_2 \cdots w_n$ and each $w_i \in \Sigma$. We further extend f to operate on languages by defining $f(A) = \{f(w) |\ w \in A\}$, for any language A.

a. Show, by giving a formal construction, that the class of regular languages is closed under homomorphism. In other words, given a DFA M that recognizes B and a homomorphism f, construct a finite automaton M' that recognizes $f(B)$. Consider the machine M' that you constructed. Is it a DFA in every case?

b. Show, by giving an example, that the class of non-regular languages is not closed under homomorphism.

1.61** Let the ***rotational closure of language A be $RC(A) = \{yx |\ xy \in A\}$.

a. Show that for any language A, we have $RC(A) = RC(RC(A))$.

b. Show that the class of regular languages is closed under rotational closure.

1.62 Let $\Sigma = \{0, 1, +, =\}$ and

$$ADD = \{x{=}y{+}z |\ x, y, z \text{ are binary integers, and } x \text{ is the sum of } y \text{ and } z\}.$$

Show that ADD is not regular.

***1.63** If A is a set of natural numbers and k is a natural number greater than 1, let

$$B_k(A) = \{w |\ w \text{ is the representation in base } k \text{ of some number in } A\}.$$

Here, we do not allow leading 0s in the representation of a number. For example, $B_2(\{3,5\}) = \{11, 101\}$ and $B_3(\{3,5\}) = \{10, 12\}$. Give an example of a set A for which $B_2(A)$ is regular but $B_3(A)$ is not regular. Prove that your example works.

***1.64** If A is any language, let $A_{\frac{1}{2}-}$ be the set of all first halves of strings in A so that

$$A_{\frac{1}{2}-} = \{x |\ \text{for some } y,\ |x| = |y| \text{ and } xy \in A\}.$$

Show that if A is regular, then so is $A_{\frac{1}{2}-}$.