



Introduction to Automata Theory, Languages, and Computation

Solutions for Chapter 2

Revised 9/6/01.

Solutions for Section 2.2

Exercise 2.2.1(a)

States correspond to the eight combinations of switch positions, and also must indicate whether the previous roll came out at D , i.e., whether the previous input was accepted. Let 0 represent a position to the left (as in the diagram) and 1 a position to the right. Each state can be represented by a sequence of three 0's or 1's, representing the directions of the three switches, in order from left to right. We follow these three bits by either a indicating it is an accepting state or r , indicating rejection. Of the 16 possible states, it turns out that only 13 are accessible from the initial state, 000r. Here is the transition table:

	A	B
->000r	100r	011r
*000a	100r	011r
*001a	101r	000a
010r	110r	001a
*010a	110r	001a
011r	111r	010a
100r	010r	111r
*100a	010r	111r
101r	011r	100a
*101a	011r	100a
110r	000a	101a
*110a	000a	101a
111r	001a	110a

Exercise 2.2.2

In what follows, we use \hat{d} for "delta-hat." The statement to be proved is $\hat{d}(q,xy) = \hat{d}(\hat{d}(q,x),y)$, and we proceed by induction on the length of y .

Basis: If $y = \epsilon$, then the statement is $\hat{d}(q,x) = \hat{d}(\hat{d}(q,x),\epsilon)$. This statement follows from the basis in the definition of \hat{d} . Note that in applying this definition, we must treat $\hat{d}(q,x)$ as if it were just a state, say p . Then, the statement to be proved is $p = \hat{d}(p,\epsilon)$, which is easy to recognize as the basis in the definition of \hat{d} .

Induction: Assume the statement for strings shorter than y , and break $y = za$, where a is the last symbol of y . The steps converting $\hat{d}(\hat{d}(q,x),y)$ to $\hat{d}(q,xy)$ are summarized in the following table:

Expression	Reason
$\hat{d}(\hat{d}(q,x),y)$	Start
$\hat{d}(\hat{d}(q,x),za)$	$y=za$ by assumption
$\delta(\hat{d}(\hat{d}(q,x),z),a)$	Definition of \hat{d} , treating $\hat{d}(q,x)$ as a state
$\delta(\hat{d}(q,xz),a)$	Inductive hypothesis
$\hat{d}(q,xza)$	Definition of \hat{d}
$\hat{d}(q,xy)$	$y=za$

Exercise 2.2.4(a)

The intuitive meanings of states A , B , and C are that the string seen so far ends in 0, 1, or at least 2 zeros.

	0	1
$\rightarrow A$	B	A
B	C	A
*C	C	A

Exercise 2.2.6(a)

The trick is to realize that reading another bit either multiplies the number seen so far by 2 (if it is a 0), or multiplies by 2 and then adds 1 (if it is a 1). We don't need to remember the entire number seen --- just its remainder when divided by 5. That is, if we have any number of the form $5a+b$, where b is the remainder, between 0 and 4, then $2(5a+b) = 10a+2b$. Since $10a$ is surely divisible by 5, the remainder of $10a+2b$ is the same as the remainder of $2b$ when divided by 5. Since b is 0, 1, 2, 3, or 4, we can tabulate the answers easily. The same idea holds if we want to consider what happens to $5a+b$ if we multiply by 2 and add 1.

The table below shows this automaton. State qi means that the input seen so far has remainder i when divided by 5.

	0	1
->*q0	q0	q1
q1	q2	q3
q2	q4	q0
q3	q1	q2
q4	q3	q4

There is a small matter, however, that this automaton accepts strings with leading 0's. Since the problem calls for accepting only those strings that begin with 1, we need an additional state s , the start state, and an additional "dead state" d . If, in state s , we see a 1 first, we act like $q0$; i.e., we go to state $q1$. However, if the first input is 0, we should never accept, so we go to state d , which we never leave. The complete automaton is:

	0	1
->s	d	q1
*q0	q0	q1
q1	q2	q3
q2	q4	q0
q3	q1	q2
q4	q3	q4
d	d	d

Exercise 2.2.9

Part (a) is an easy induction on the length of w , starting at length 1.

Basis: $|w| = 1$. Then $dhat(q_0, w) = dhat(q_f, w)$, because w is a single symbol, and $dhat$ agrees with $delta$ on single symbols.

Induction: Let $w = za$, so the inductive hypothesis applies to z . Then $dhat(q_0, w) = dhat(q_0, za) = delta(dhat(q_0, z), a) = delta(dhat(q_f, z), a)$ [by the inductive hypothesis] $= dhat(q_f, za) = dhat(q_f, w)$.

For part (b), we know that $dhat(q_0, x) = q_f$. Since $x \in \epsilon$, we know by part (a) that $dhat(q_f, x) = q_f$. It is then a simple induction on k to show that $dhat(q_0, x^k) = q_f$.

Basis: For $k=1$ the statement is given.

Induction: Assume the statement for $k-1$; i.e., $\text{dhat}(q_0, x^{k-1}) = q_f$. Using Exercise 2.2.2, $\text{dhat}(q_0, x^k) = \text{dhat}(\text{dhat}(q_0, x^{k-1}), x) = \text{dhat}(q_f, x)$ [by the inductive hypothesis] $= q_f$ [by (a)].

Exercise 2.2.10

The automaton tells whether the number of 1's seen is even (state A) or odd (state B), accepting in the latter case. It is an easy induction on $|w|$ to show that $\text{dh}(A, w) = A$ if and only if w has an even number of 1's.

Basis: $|w| = 0$. Then w , the empty string surely has an even number of 1's, namely zero 1's, and $\text{dhat}(A, w) = A$.

Induction: Assume the statement for strings shorter than w . Then $w = za$, where a is either 0 or 1.

Case 1: $a = 0$. If w has an even number of 1's, so does z . By the inductive hypothesis, $\text{dhat}(A, z) = A$. The transitions of the DFA tell us $\text{dhat}(A, w) = A$. If w has an odd number of 1's, then so does z . By the inductive hypothesis, $\text{dhat}(A, z) = B$, and the transitions of the DFA tell us $\text{dhat}(A, w) = B$. Thus, in this case, $\text{dhat}(A, w) = A$ if and only if w has an even number of 1's.

Case 2: $a = 1$. If w has an even number of 1's, then z has an odd number of 1's. By the inductive hypothesis, $\text{dhat}(A, z) = B$. The transitions of the DFA tell us $\text{dhat}(A, w) = A$. If w has an odd number of 1's, then z has an even number of 1's. By the inductive hypothesis, $\text{dhat}(A, z) = A$, and the transitions of the DFA tell us $\text{dhat}(A, w) = B$. Thus, in this case as well, $\text{dhat}(A, w) = A$ if and only if w has an even number of 1's.

Solutions for Section 2.3

Exercise 2.3.1

Here are the sets of NFA states represented by each of the DFA states A through H: $A = \{p\}$; $B = \{p, q\}$; $C = \{p, r\}$; $D = \{p, q, r\}$; $E = \{p, q, s\}$; $F = \{p, q, r, s\}$; $G = \{p, r, s\}$; $H = \{p, s\}$.

	0	1
->A	B	A
B	D	C
C	E	A
D	F	C
*E	F	G
*F	F	G
*G	E	H

*H	E	H
----	---	---

Exercise 2.3.4(a)

The idea is to use a state q_i , for $i = 0, 1, \dots, 9$ to represent the idea that we have seen an input i and guessed that this is the repeated digit at the end. We also have state q_s , the initial state, and q_f , the final state. We stay in state q_s all the time; it represents no guess having been made. The transition table:

	0	1	...	9
-> q_s	{ q_s, q_0 }	{ q_s, q_1 }	...	{ q_s, q_9 }
q_0	{ q_f }	{ q_0 }	...	{ q_0 }
q_1	{ q_1 }	{ q_f }	...	{ q_1 }
...
q_9	{ q_9 }	{ q_9 }	...	{ q_f }
* q_f	{}	{}	...	{}

Solutions for Section 2.4

Exercise 2.4.1(a)

We'll use q_0 as the start state. q_1 , q_2 , and q_3 will recognize abc ; q_4 , q_5 , and q_6 will recognize abd , and q_7 through q_{10} will recognize $aacd$. The transition table is:

	a	b	c	d
-> q_0	{ q_0, q_1, q_4, q_7 }	{ q_0 }	{ q_0 }	{ q_0 }
q_1	{}	{ q_2 }	{}	{}
q_2	{}	{}	{ q_3 }	{}
* q_3	{}	{}	{}	{}
q_4	{}	{ q_5 }	{}	{}
q_5	{}	{}	{}	{ q_6 }
* q_6	{}	{}	{}	{}
q_7	{ q_8 }	{}	{}	{}
q_8	{}	{}	{ q_9 }	{}
q_9	{}	{}	{}	{ q_{10} }
* q_{10}	{}	{}	{}	{}

Exercise 2.4.2(a)

The subset construction gives us the following states, each representing the subset of the NFA states indicated: $A = \{q_0\}$; $B = \{q_0, q_1, q_4, q_7\}$; $C = \{q_0, q_1, q_4, q_7, q_8\}$; $D = \{q_0, q_2, q_5\}$; $E = \{q_0, q_9\}$; $F = \{q_0, q_3\}$; $G = \{q_0, q_6\}$; $H = \{q_0, q_{10}\}$. Note that F , G and H can be combined into one accepting state, or we can use these three state to signal the recognition of abc , abd , and $aacd$, respectively.

	a	b	c	d
->A	B	A	A	A
B	C	D	A	A
C	C	D	E	A
D	B	A	F	G
E	B	A	A	H
*F	B	A	A	A
*G	B	A	A	A
*H	B	A	A	A

Solutions for Section 2.5

Exercise 2.5.1

For part (a): the closure of p is just $\{p\}$; for q it is $\{p, q\}$, and for r it is $\{p, q, r\}$.

For (b), begin by noticing that a always leaves the state unchanged. Thus, we can think of the effect of strings of b 's and c 's only. To begin, notice that the only ways to get from p to r for the first time, using only b , c , and epsilon-transitions are bb , bc , and c . After getting to r , we can return to r reading either b or c . Thus, every string of length 3 or less, consisting of b 's and c 's only, is accepted, with the exception of the string b . However, we have to allow a 's as well. When we try to insert a 's in these strings, yet keeping the length to 3 or less, we find that every string of a 's b 's, and c 's with at most one a is accepted. Also, the strings consisting of one c and up to 2 a 's are accepted; other strings are rejected.

There are three DFA states accessible from the initial state, which is the epsilon closure of p , or $\{p\}$. Let $A = \{p\}$, $B = \{p, q\}$, and $C = \{p, q, r\}$. Then the transition table is:

	a	b	c
->A	A	B	C
B	B	C	C
*C	C	C	C