

Exercise 8

Brandon Hosley

Mike Davis

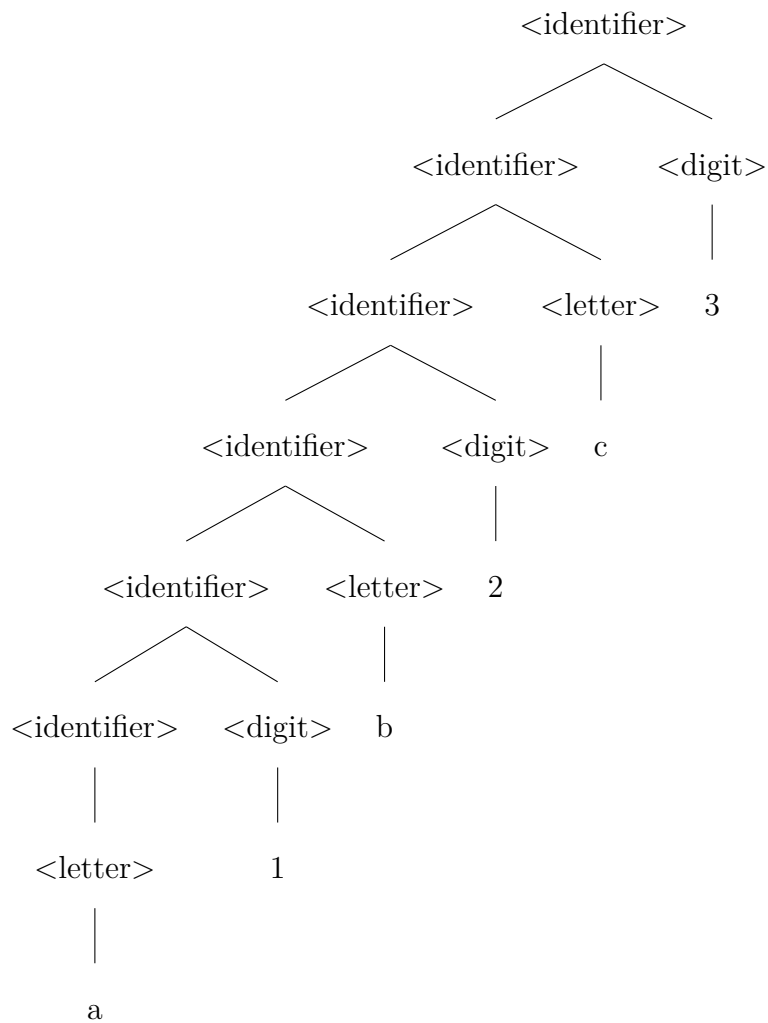
Exercise 8

7.3

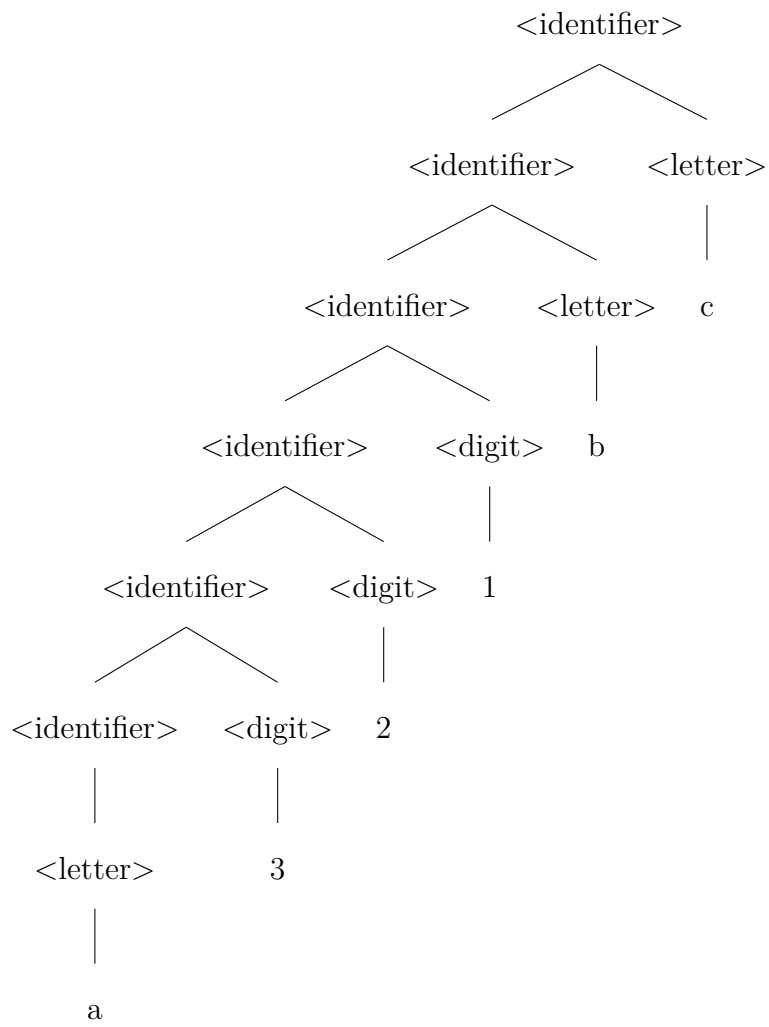
Derive the following strings with the grammar of Figure 7.1 and draw the corresponding syntax tree:

$N = \{ \langle \text{identifier} \rangle, \langle \text{letter} \rangle, \langle \text{digit} \rangle \}$
 $T = \{ a, b, c, 1, 2, 3 \}$
 $P = \text{the productions}$
 1. $\langle \text{identifier} \rangle \rightarrow \langle \text{letter} \rangle$
 2. $\langle \text{identifier} \rangle \rightarrow \langle \text{identifier} \rangle \langle \text{letter} \rangle$
 3. $\langle \text{identifier} \rangle \rightarrow \langle \text{identifier} \rangle \langle \text{digit} \rangle$
 4. $\langle \text{letter} \rangle \rightarrow a$
 5. $\langle \text{letter} \rangle \rightarrow b$
 6. $\langle \text{letter} \rangle \rightarrow c$
 7. $\langle \text{digit} \rangle \rightarrow 1$
 8. $\langle \text{digit} \rangle \rightarrow 2$
 9. $\langle \text{digit} \rangle \rightarrow 3$
 $S = \langle \text{identifier} \rangle$

b. a1b2c3



c. a321bc

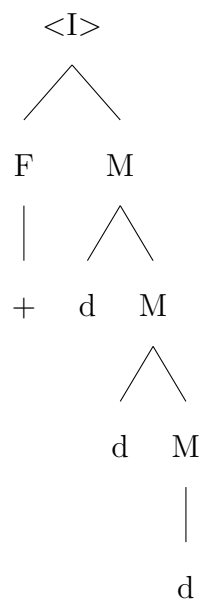


7.4

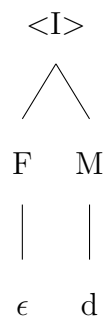
Derive the following strings with the grammar of Figure 7.2 and draw the corresponding syntax tree:

$N = \{I, F, M\}$
 $T = \{+, -, d\}$
 $P =$ the productions
 1. $I \rightarrow FM$
 2. $F \rightarrow +$
 3. $F \rightarrow -$
 4. $F \rightarrow \epsilon$
 5. $M \rightarrow dM$
 6. $M \rightarrow d$
 $S = I$

b. $+ddd$



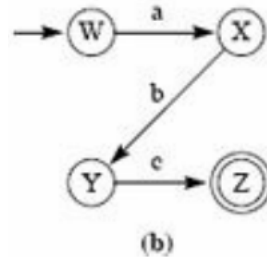
c. d



7.11

For each of the machines shown in Figure 7.48, **(a)** state whether the FSM is deterministic or nondeterministic and **(b)** identify any states that are inaccessible.

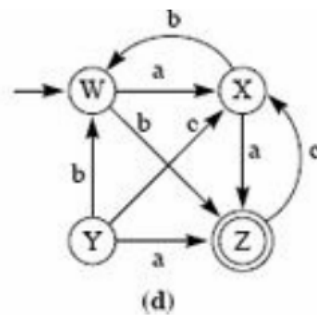
b.



(a) This FSM is deterministic.

(b) All states are accessible in this FSM.

d.



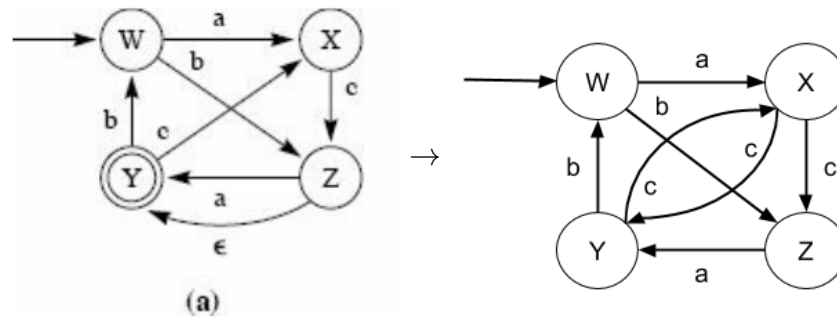
(a) This FSM is deterministic.

(b) State (Y) is inaccessible in this FSM.

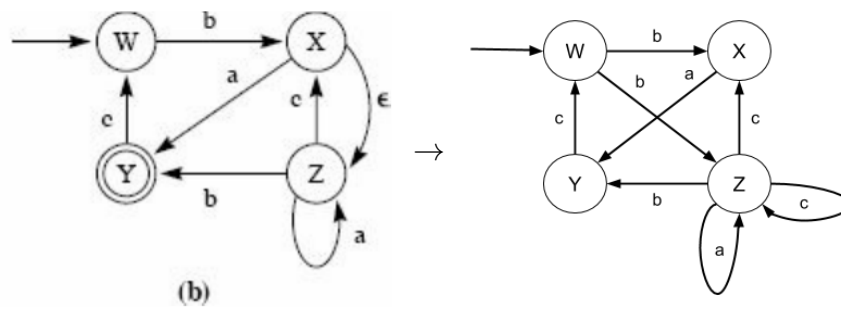
7.12

Remove the empty transitions to produce the equivalent machine for each of the finite state machines in Figure 7.49.

a.



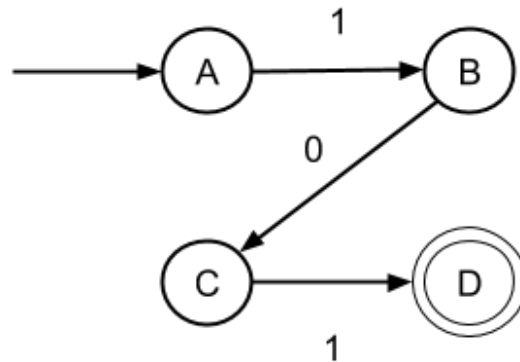
b.



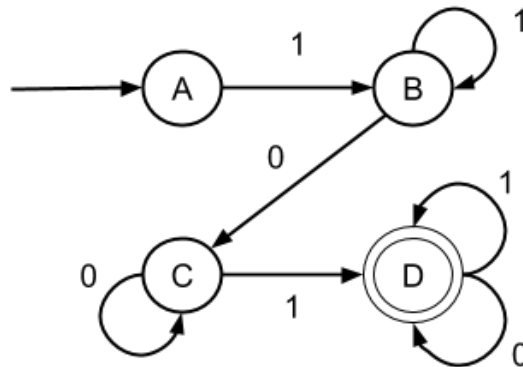
7.13

Draw a deterministic FSM that recognizes strings of 1's and 0's specified by each of the following criteria. Each FSM should reject any characters that are not 0 or 1.

- a. The string of three characters, 101.



- d. All strings of arbitrary length that contain a 101 at least once anywhere. For example, the FSM should accept all the strings mentioned in parts (a), (b), and (c) as well as strings such as 11100001011111100111.



References

Warford, J. (2009). *Computer systems* (4th ed.). Jones and Bartlett.