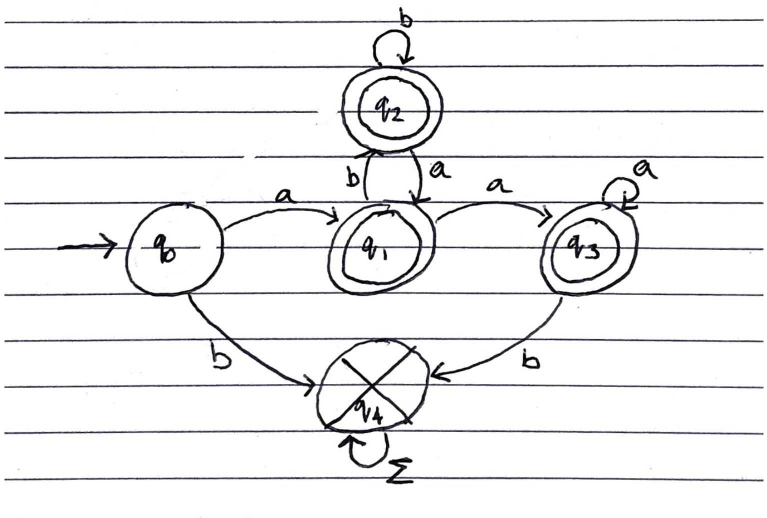
Stefanie Shidoosh

CS 181: HW #3

1. Let alphabet Σ = {a, b}. Show a DFA for the following language:
L = {all strings in Σ* which begin with symbol a and do not contain aab as a substring}
You may represent your DFA in any format we have used. It does not need to be fully specified, but it must be clear. Be sure to clearly indicate your initial state and final states.



q_0 is the initial state. q_1, q_2, q_3 are the accepting states. q_4 is a dead state.

Stefanie Shidoosh

2. Let $\Sigma = \{0, 1\}$, and let A, B, N, X be four different languages over Σ, each of which contains an infinite number of words. Let A & B be FSLs. Let N be a non-FSL. Whether X is a FSL or a non-FSL will depend on additional assumptions provided below for each question. Classify each of the following languages as a FSL, a Non-FSL, or Unknown (i.e., it is impossible to determine from only the given information.). Briefly justify your answer.

a. $A \cap B$

FSL. The intersection of an FSL with another FSL will result in an FSL, as the intersection operation is closed with respect to FSLs.

b. $B \cap N$

Unknown. The intersection of an FSL and a non-FSL is inconclusive.

Consider the regular languages $\varnothing$, Σ^*. Let L be any non-regular language over Σ.

$\Sigma^* \cap L = L$ is non-regular. However, $\varnothing \cap L = \varnothing$ is regular.

c. $A \cup N$

Unknown. The union of an FSL and a non-FSL is inconclusive.

Consider the regular languages $\varnothing$, Σ^*. Let L be any non-regular language over Σ.

$\varnothing \cup L = L$ is non-regular. However, $\Sigma^* \cup L = \Sigma^*$ is regular.

d. X, assuming we add the assumption just for this part of the question that $X \cup A$ is not a FSL.

Unknown. It is possible that X or A could be the non-FSL, or both X and A are non-FSLs.

e. X, assuming we add the assumption just for this part of the question that X is a subset of B.

Unknown. FSLs are not closed under subset and proper subset operations.

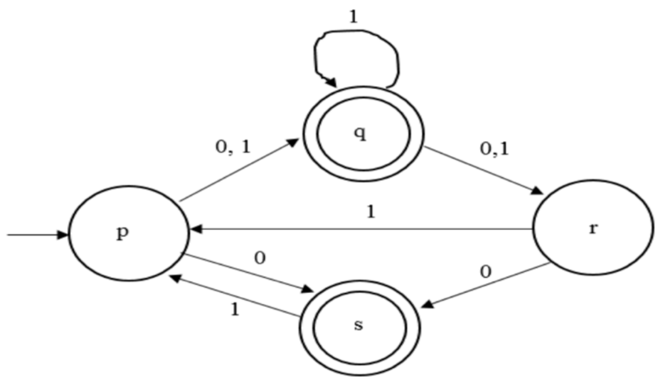
Let L be any non-regular language on Σ. Then L is a subset of Σ^*, and Σ^* is regular.

However, Σ^* is a subset of Σ^*, thus this subset is regular.

Thus, we cannot conclude that the subset of a regular language guaranteed to be a FSL or non-FSL.

Stefanie Shidoosh

3. Let N be the NFA over alphabet $\Sigma = \{0, 1\}$ specified by the following transition function state transition diagram, where p is the initial state and q & s are the Final/Accepting states:



Note that N is not fully specified; so this diagram relies on the blocking convention for all unspecified transitions of N. Convert N to a DFA using the procedure in the proof of Sipser Theorem 1.39, pp 55-56. Be sure to show your work and clearly specify your final result.

Let N = $(Q, \Sigma, \delta, q_0, F)$

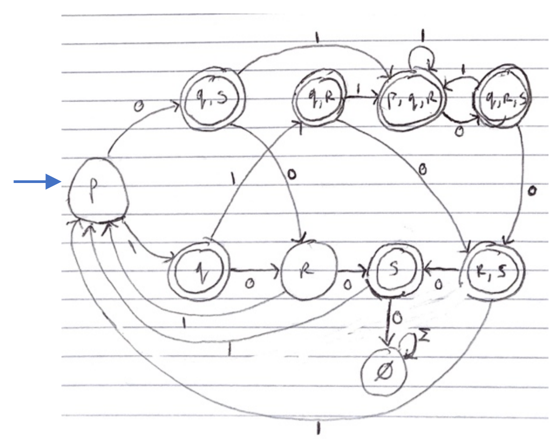
We construct a DFA M $=(Q', \Sigma, \delta', q_0', F')$

$Q' = P(Q)$

$\Sigma = \{0, 1\}$

δ':

	0	1
{p}	{q,s}	{q}
{q}	{r}	{q,r}
{r}	{s}	{p}
{s}	{}	{p}
{q,s}	{r}	{p,q,r}
{q,r}	{r,s}	{p,q,r}
{r,s}	{s}	{p}
{p,q,r}	{q,r,s}	{p,q,r}
{q,r,s}	{r,s}	{p,q,r}
{}	{}	{}



I determined this transition by using Theorem 1.39: The transition is determined by defining the consequent state(s) on a given input. If there is more than one state possible for a single input, union those states to a single, distinct state. In δ', I have omitted the states that do not have an arrow pointing to it for sake of simplicity.

$q_0' = \{q_0\} = \{p\}$

$F' = \{R \in Q' \mid R$ contains an accept state of N$\} = \{\{q\}, \{s\}, \{q,s\}, \{q,r\}, \{r,s\}, \{p,q,r\}, \{q,r,s\}\}$.

Stefanie Shidoosh

4. Consider the following GNFA over alphabet $\Sigma = \{a, b\}$ from Sipser:

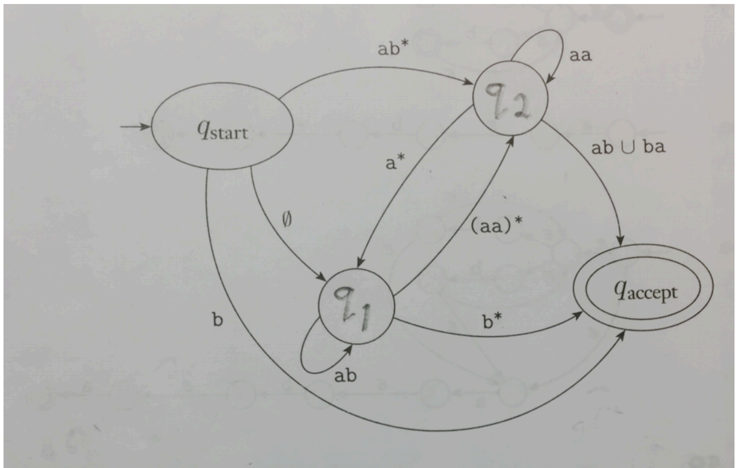


FIGURE 1.61
A generalized nondeterministic finite automaton

Determine whether each of the following strings is in the language of the GNFA:

a. ab

This string would not be in the language of this GNFA. There are two possibilities:

1. q_{start} → q_2, via just a.
2. q_{start}→ q_2, via ab.

In both scenarios, there is no available transition from q_2 to any other state:

1. only input left is b, which is not available on the transitions from q_2, and
2. no input left to traverse next transition, no epsilon move.

Therefore, blocking is assumed and a dead state is reached.

b. aabb

This string would not be in the language of this GNFA. There are no available transitions from q_{start} that expresses this string. The available transitions associated with q_{start} are ab*, b, Ø, none of which allows for aabb, thus blocking is assumed and a dead state is reached.

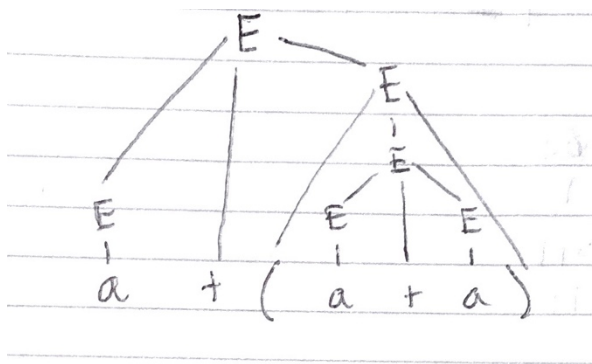
Stefanie Shidoosh

5. Let $\Sigma = \{a, b\}$. Give a regular expression for the following language from last week:
L5 = {w in $\Sigma +$ | $|w| > 3$ and the same symbol appears in the position 4th from the end and in the position 2nd from the end}
E.g., bbabbba and abab are in L4, but abbb is not. Part of your score will be based on demonstrating effective use of the regular expression model.

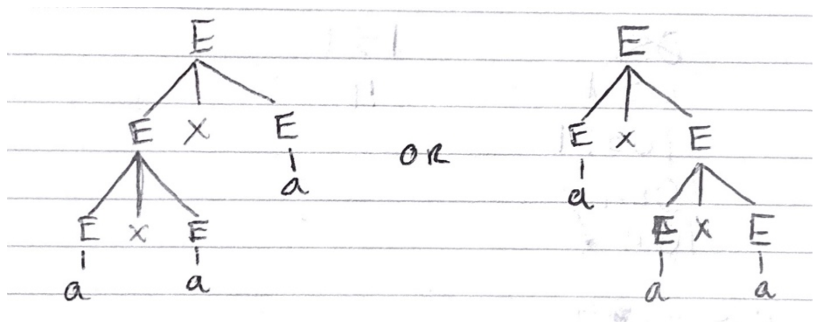
$$(\Sigma^*a\Sigma a\Sigma) \cup (\Sigma^*b\Sigma b\Sigma)$$

Stefanie Shidoosh

6. Let G5 = ({E}, Σ, R, E) be the CFG over alphabet Σ= { a, +, x, (,) } with rule set,
R: E -> E + E | E x E | (E) | a
(Compare to the grammar on pages 107-108 of Sipser.)
a. Show the parse tree in G5 for the following string from L(G5): a + (a + a).



b. Show two different parse trees in G5 for the following string: a x a x a .



c. Show the two left-most derivations in G5 (using “E” instead of) corresponding to the two parse trees at the top of page 108:

108 CHAPTER 2 / CONTEXT-FREE LANGUAGES

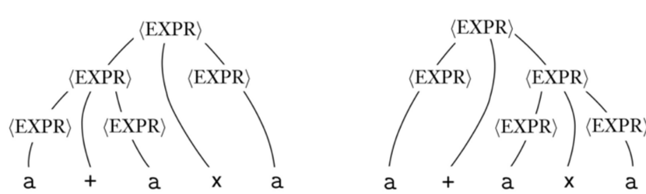


FIGURE **2.6**
The two parse trees for the string a+axa in grammar G_5

Be sure to indicate which left-most derivation corresponds to which tree.
The tree on the left: E => E x E=> E+E x E => a + E x E => a + a x E => a + a x a
The tree on the right: E => E + E => a + E => a + E x E => a + a x E => a + a x a

Stefanie Shidoosh

7. Let $\Sigma = \{0, 1\}$.
Show a CFG for the following language:
L7 = $\{xy \in \Sigma + \mid |x| = |y|$ and y differs from x^R in exactly one position$\}$
Be sure to clearly indicate your set of Variables, your start variable, and your rules.

G = (V, Σ, R, S)

V = {S, X, Y}

Σ = {0, 1}

R, the set of rules, is:

S → 1X1 | 0X0 | 1Y0 | 0Y1

X → 0Y1 | 1Y0 | 1S1 | 0S0

Y → 0Y0 | 1Y1 | ε

S is the start variable.

Stefanie Shidoosh

8. Let Σ = { 0, 1 }, and let
L8 = { w ∈ Σ* | w contains an equal number of 01 substrings and 10 substrings }
Fact: L8 is either a FSL or a non-FSL CFL.
Determine whether L8 is a FSL or a non-FSL CFL.
If it is a FSL, show that using any representation we have used in class. (E.g., DFA, NFA, Regular Expression.)
If it is not a FSL, show that using the Pumping Lemma and give a CFG for it.
In either case, briefly describe how your model (DFA, NFA, Regular Expression, CFG) works. Do not jump to any conclusions about the answer just because of the way the question is written.

> L8 is a FSL: This DFA model works by keeping track of the occurrences of each substring 01 and 10, respectively. That is, for every substring 01 traversed, the substring 10 must be traversed at some point in succession in order to reach an accept state. Also, for every substring 10 traversed, the substring 01 must be traversed at some point in succession in order to reach an accept state. Lastly, the empty string is in L8 and thus the start state is an accepting state.