Gabriel Arnell

Lab 2

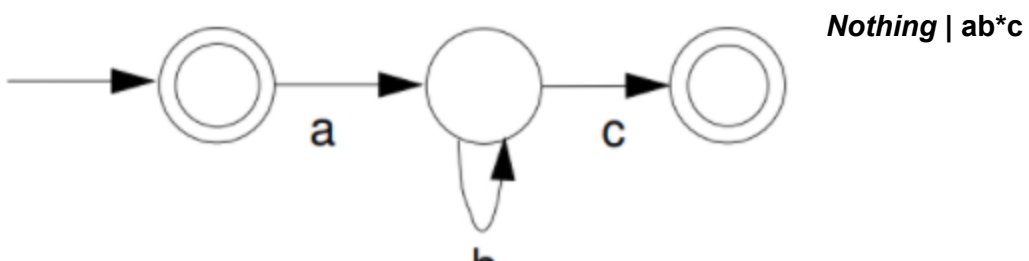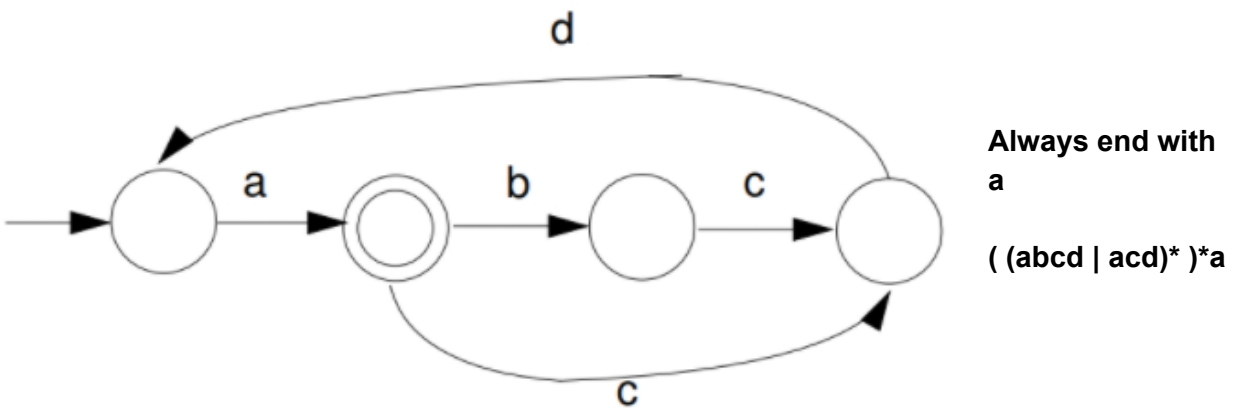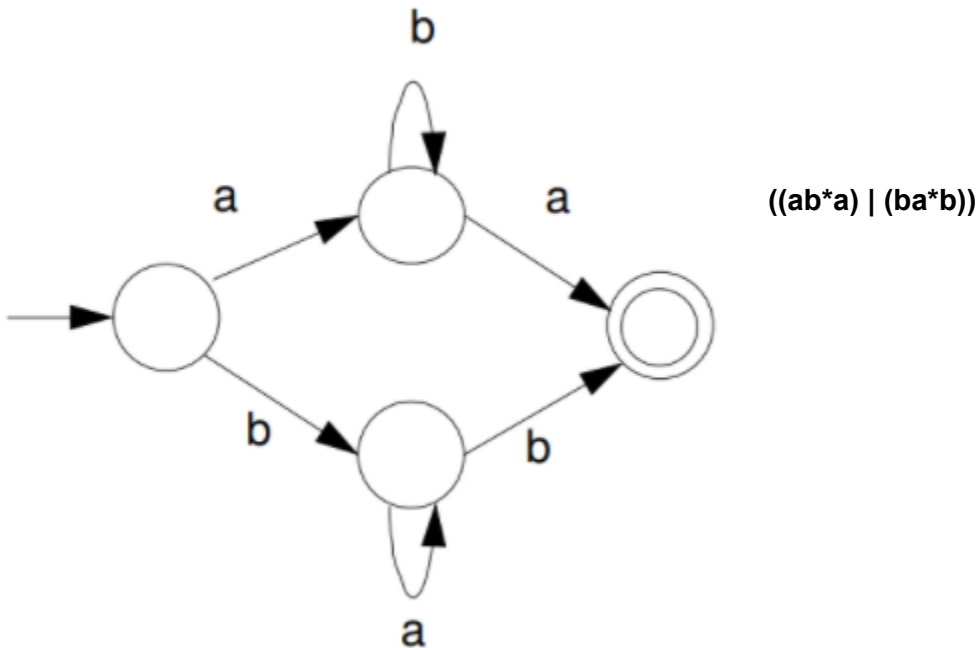**Crafting a Compiler 3.3**

Write regular expressions that define the strings recognized by the FAs in figure 3.33 on page 107.

b

a   a   ((ab*a) | (ba*b))

b   b

a

d

Always end with
a

a   b   c

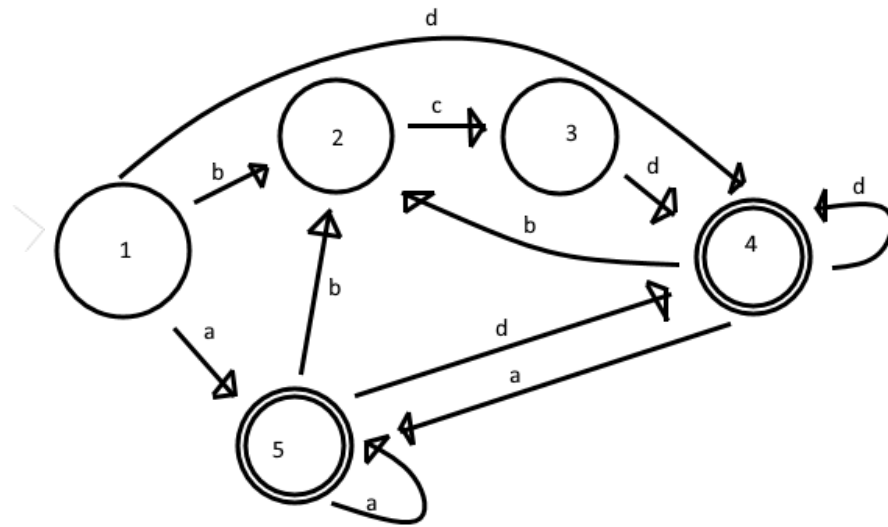((abcd | acd)*)*a

c

Nothing | ab*c
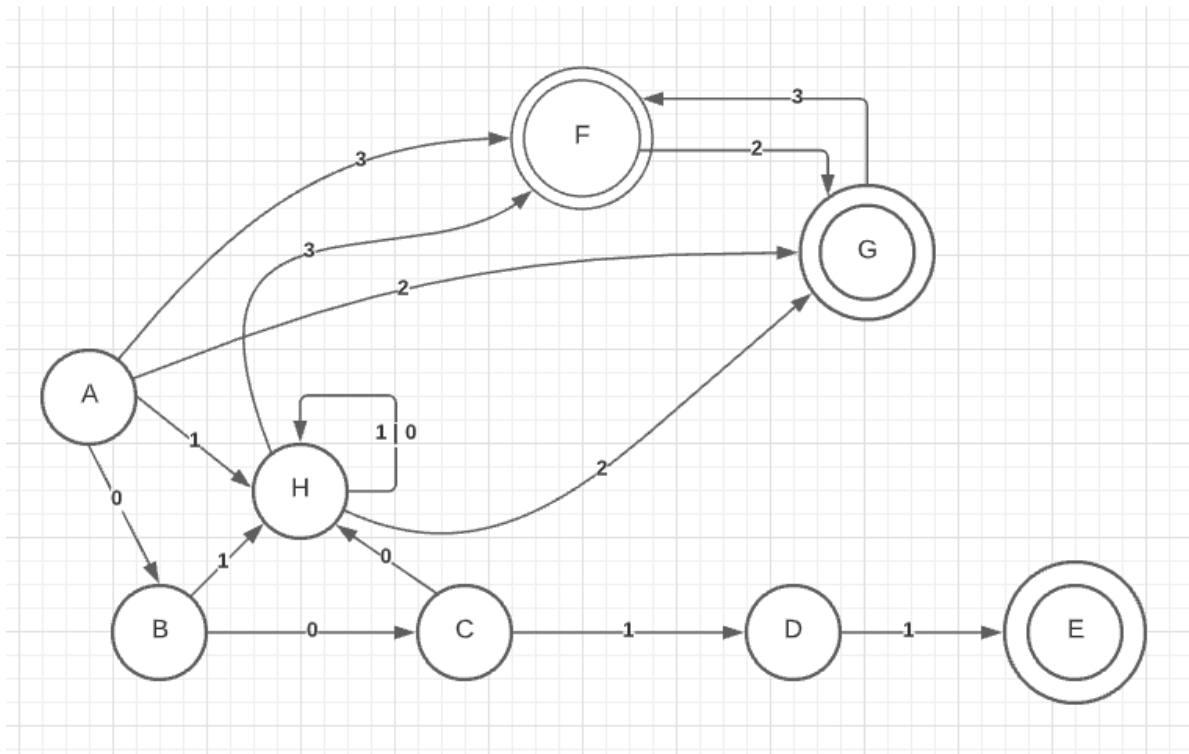
a   c

b

Crafting a Compiler 3.4
Write DFAs that recognize the tokens defined by the following regular expressions.
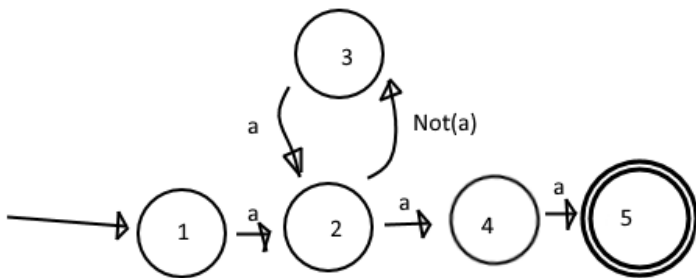**(Note: Was told in class we only had to make the NFAs**

a. $(a \mid (bc)^* d)^+$

b. $((0 \mid 1)^*(2 \mid 3)^*) \mid 0011$

c. (a Not(a))* aaa

Dragon 3.3.4

Most languages are *case sensitive*, so keywords can be written only one way, and the regular expressions describing their lexemes are very simple. However,s ome languages, like SQL, are *case insensitive*, so a keyword can be written either in lowercase or in uppercase, or any mixture of cases. Thus, the SQL keyword SELECT can also be written

select, Select, or sELECT, for instance. Show how to write a regular expression for a keyword in a case insensitive language. Illustrate the idea by writing the expression for "select" in SQL.

(s | S)(e | E)(l | L)(e | e)(c | C)(t | T)