

1 pg 106: #1

```
ID main
OPAREN
CPAREN
OCURLY
CONST
TYPE float
ID payment
EQUALS
FLOATVALUE 384.00
SEMICOLON
TYPE float
ID bal
SEMICOLON
TYPE int
ID month
EQUALS
INTVALUE 0
SEMICOLON
ID bal
EQUALS
INTVALUE
SEMICOLON
WHILE
OPAREN
ID bal
GREATERTHAN
INTVAL
CPAREN
OCURLY
ID printf
OPAREN
STRVALUE "Month: %2d Balance: %10.2f\n"
COMMA
ID month
COMMA
ID bal
CPAREN
SEMICOLON
```

ID bal
 EQUALS
 ID bal
 MINUS
 ID payment
 PLUS
 FLOATVALUE 0.015
 STAR
 ID bal
 SEMICOLON
 ID month
 EQUALS
 ID month
 PLUS
 INTVALUE 1
 SEMICOLON
 CCURLY
 CCURLY

2 pg 106 #4

a. $(a|(bc)^*d)+$

	a	b	c	d
1	4	2		4
2			3	
3		2		4
*4	4	2		4

b. $((0|1)^*(2|3)+)|0011$

	0	1	2	3
1	2	6	7	7
2	3	6	7	7
3	6	4	7	7
4	7	5	7	7
*5			7	7
6	6	6	7	7
*7			7	7

c. $(a\text{Not}(a))^*aaa$

Note: Not(a) depends on what Σ is

	a	Not(a)
1	2	
2	3	1
3	4	
*4		

3 Page 110 #19

$\text{Rev}(R)$ can be constructed from the DFA of R . Reverse the direction of all the arrows. Create a new state (start state). Then make a lambda transition from it to the end-states. Turn the old start state into an end state and the old end states into normal states. The resulting ϵ -NFA will be $\text{Rev}(R)$, and the existence of it proves its regularity.

4 107 and 111 #6 and #23

6. C-style multiline comment $/(\text{Not}(/))^*/$

23. Plurals: Does it end in an s? if the first part matches a set of words that are not regular plurals, then the ending pattern is different, with pretty unique endings. To be truly accurate, it would need to be a big set of words. For tenses, again, match a single string such that it's followed by one of the normal endings. The first group part would be the original, and the tenses from the endings.