Problem Set

9.

a.) $(((a*b)^1-1)^2>>+c)^3$

b.) $(((a*(b-1)^1)^2/c)^3 \text{mod } d)^4$

Assuming this is like regular math where you do the stuff in parentheses first.
c.) $(((a-b)^1/c)^2\&(((d*e)^3/a)^4-3)^5)^6$

d.) $((-a)^1 \text{or } ((c=d)^2 \text{and } e)^3)^4$

e.) $(((a>b)^1 \text{ xor } c)^3 \text{ or } (d<=17)^2)^4$

f.) $(-(a+b)^1)^2$

18.

I would argue that an optimizing compiler should not be able to change the order of expressions in a boolean expression. This is because in C and C++ there is something called short circuiting which means if the first part of an && equals to false or the first part of || equals True then the second half of the expression is never evaluated. With this in mind, a programmer could put something in the second half which could crash the program if it's evaluated even though it should have stopped after the first half.