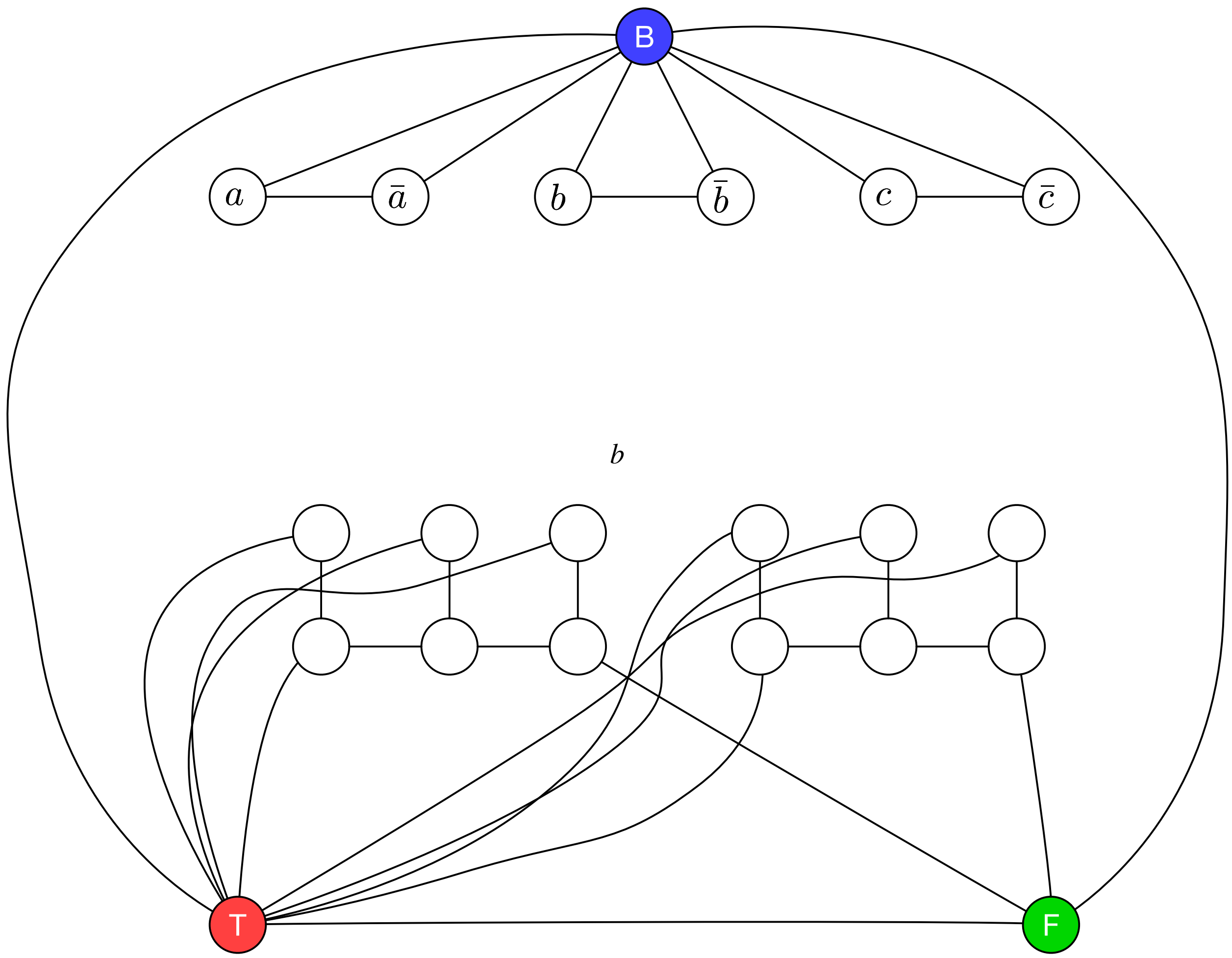
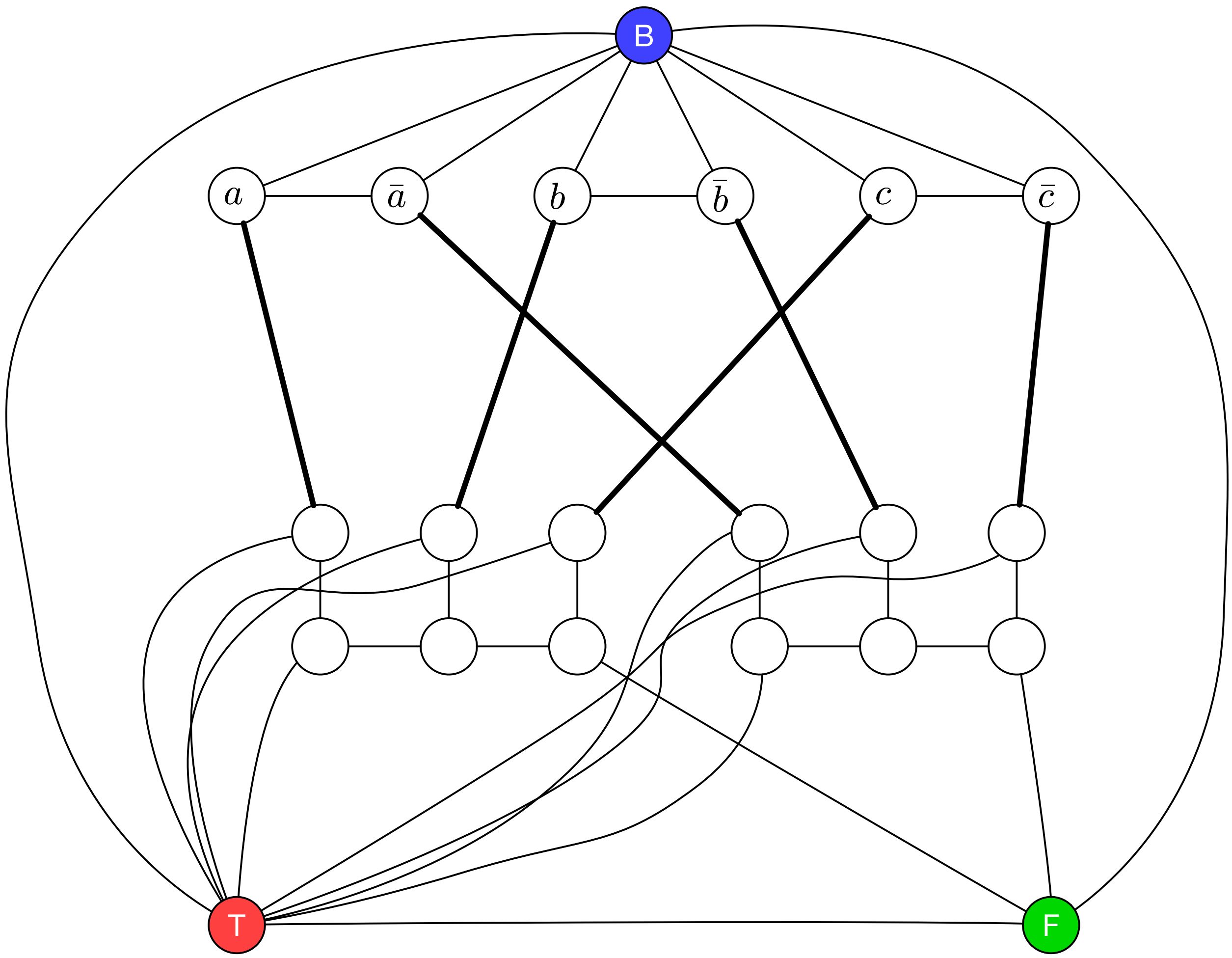
Coloring Hints

You need to combine the different components of the graph. Here are the vertices and most of the edges of a graph with two clauses:



What is lacking is connecting the literals in the second row from the top to the nodes on the third row, which is how you encode the exact 3SAT clause. For example, we have the following graph for the 3-literal conjunctive normal form

$$(a \vee b \vee c) \wedge (\neg a \vee \neg b \vee \neg c).$$



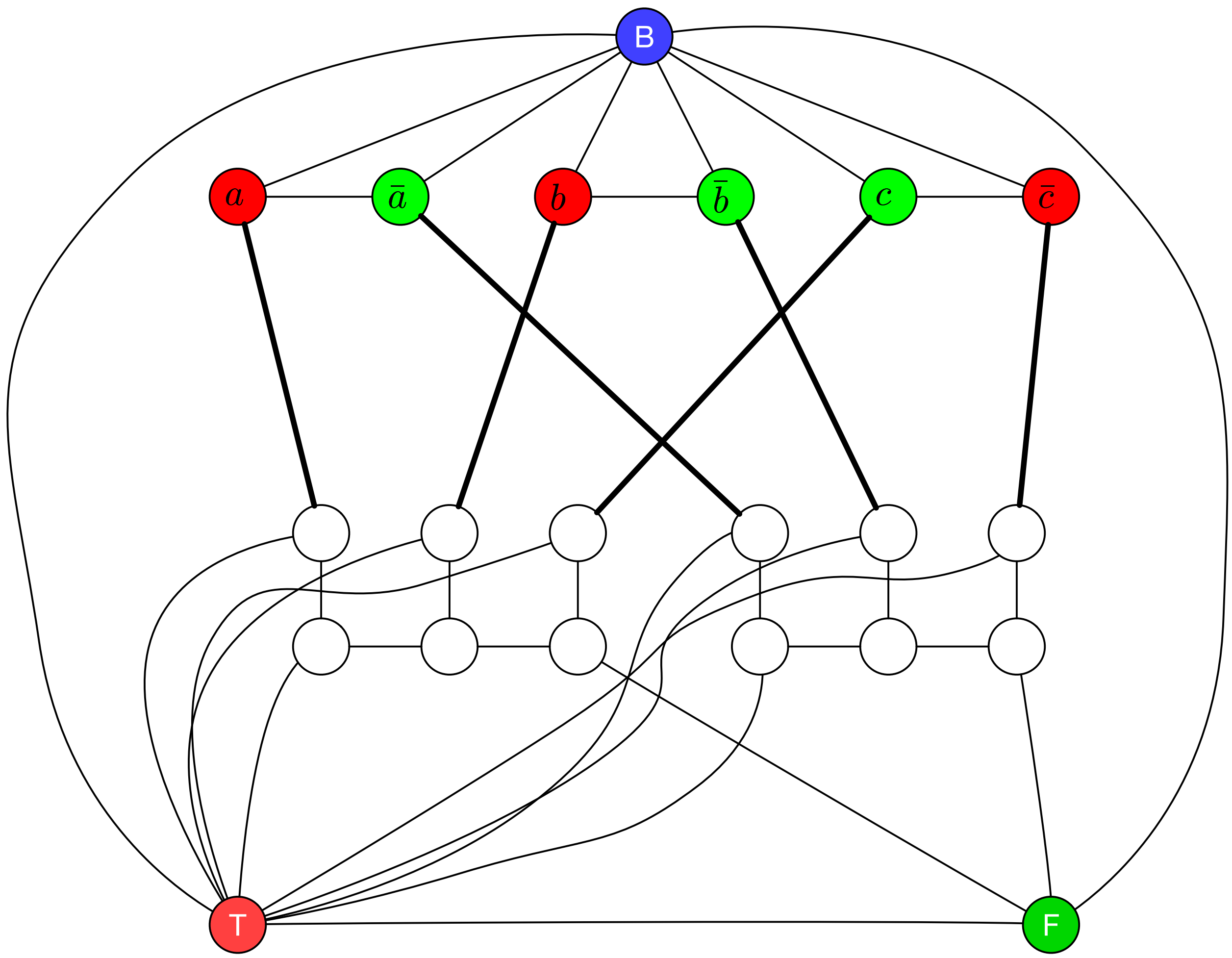
$$a \vee b \vee c \qquad \bar{a} \vee \bar{b} \vee \bar{c}$$

Let's see whether we can satisfy this. We can use a Boolean table

a	b	c	$(a \vee b \vee c)$	$(\neg a \vee \neg b \vee \neg c)$	$(a \vee b \vee c) \wedge (\neg a \vee \neg b \vee \neg c)$
0	0	0	0	1	0
0	0	1	1	1	1
0	1	0	1	1	1
0	1	1	1	1	1
1	0	0	1	1	1
1	0	1	1	1	1
1	1	0	1	1	1
1	1	1	1	0	0

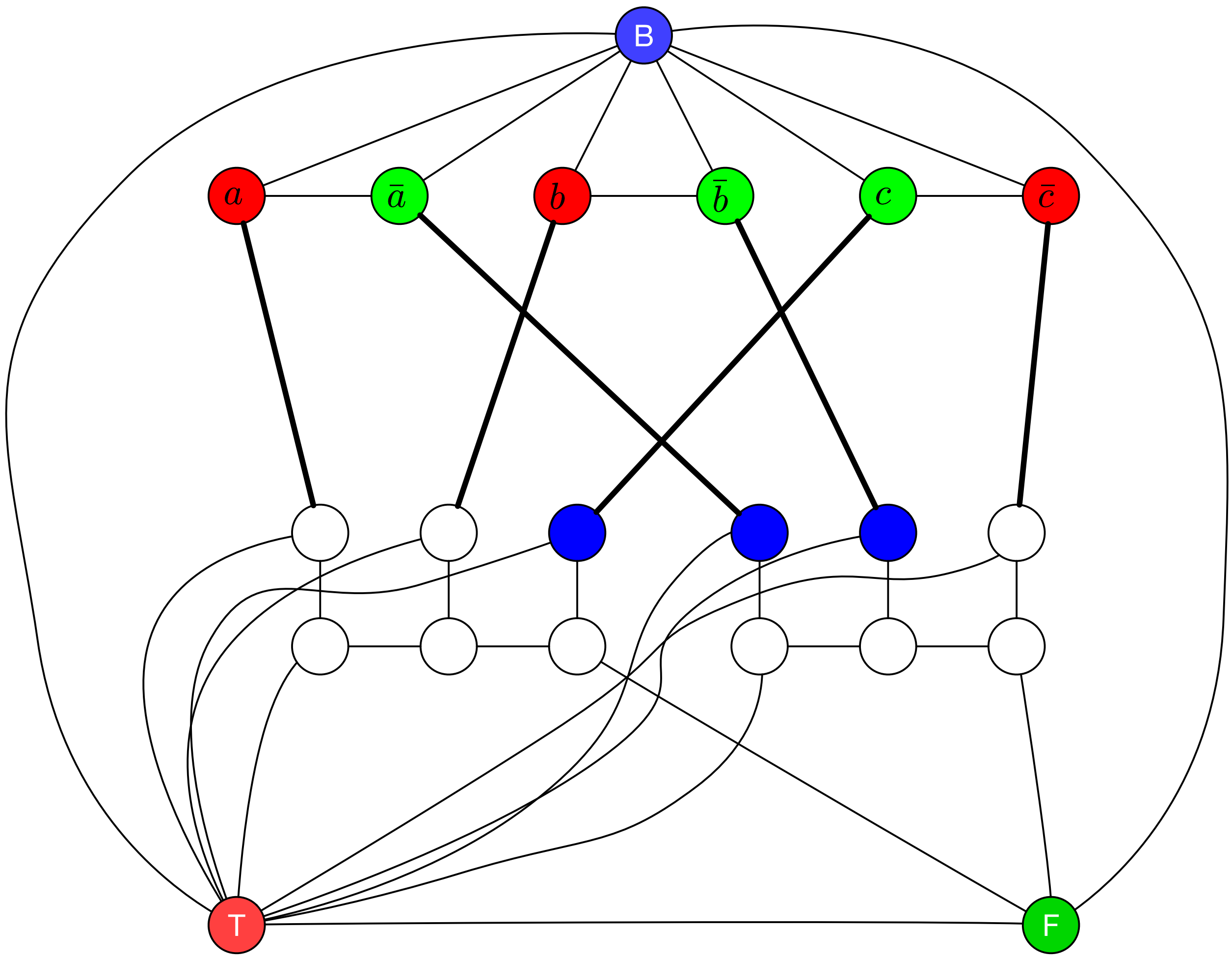
As you can see, there are lots of assignments that make this work. Let's use $a = 1$, $b = 1$, $c = 0$.

We color the literal nodes red and green accordingly. (Notice, they cannot be blue because of B being a neighbor and because the literals are connected in a triangle with B, one literal has to be red and the other one green.



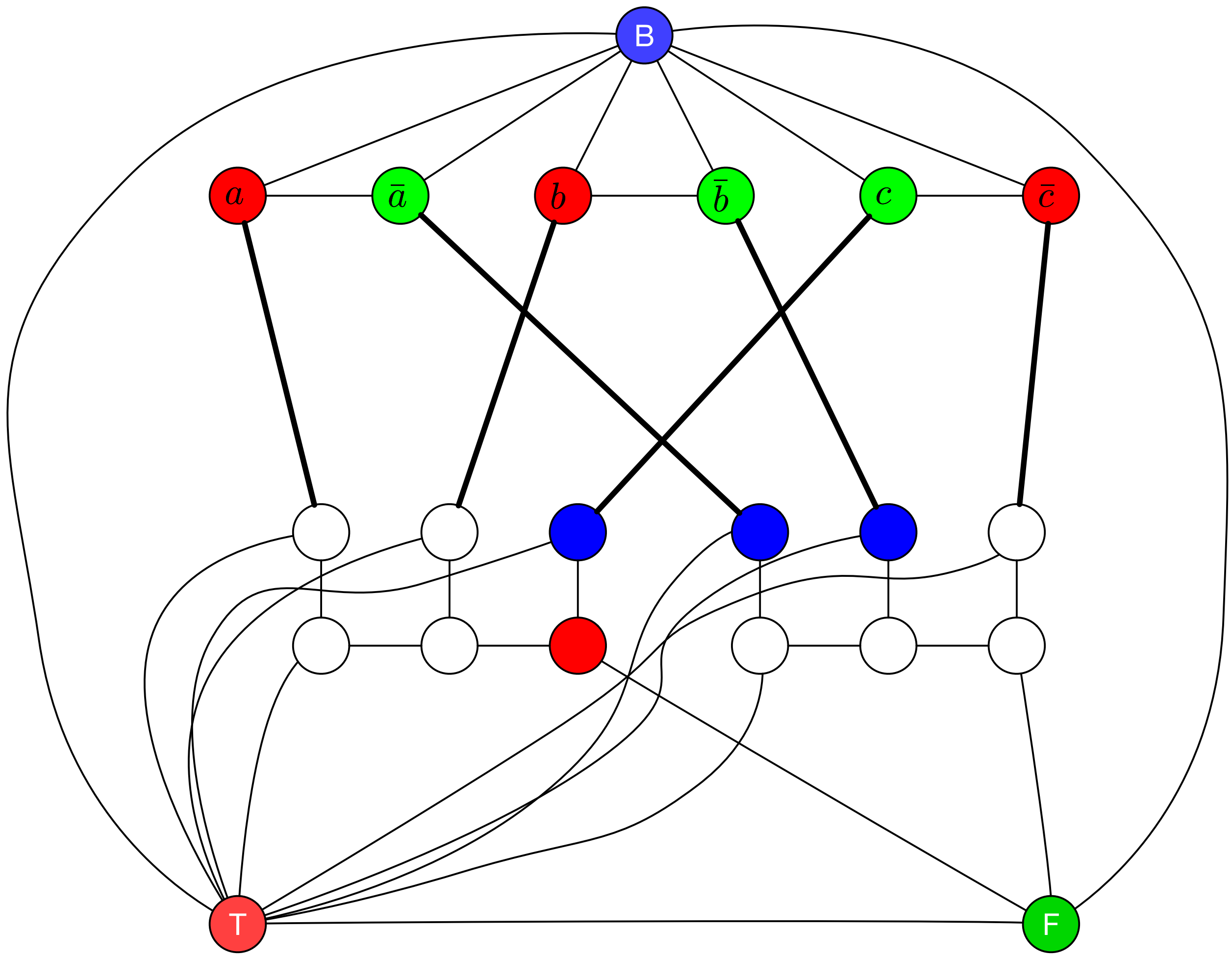
$$a \vee b \vee c \qquad \bar{a} \vee \bar{b} \vee \bar{c}$$

This forces a number of node colors in the second row. If the literal is green, then the corresponding node in the second row has to be blue, because they have T as a neighbor.



$$a \vee b \vee c \qquad \bar{a} \vee \bar{b} \vee \bar{c}$$

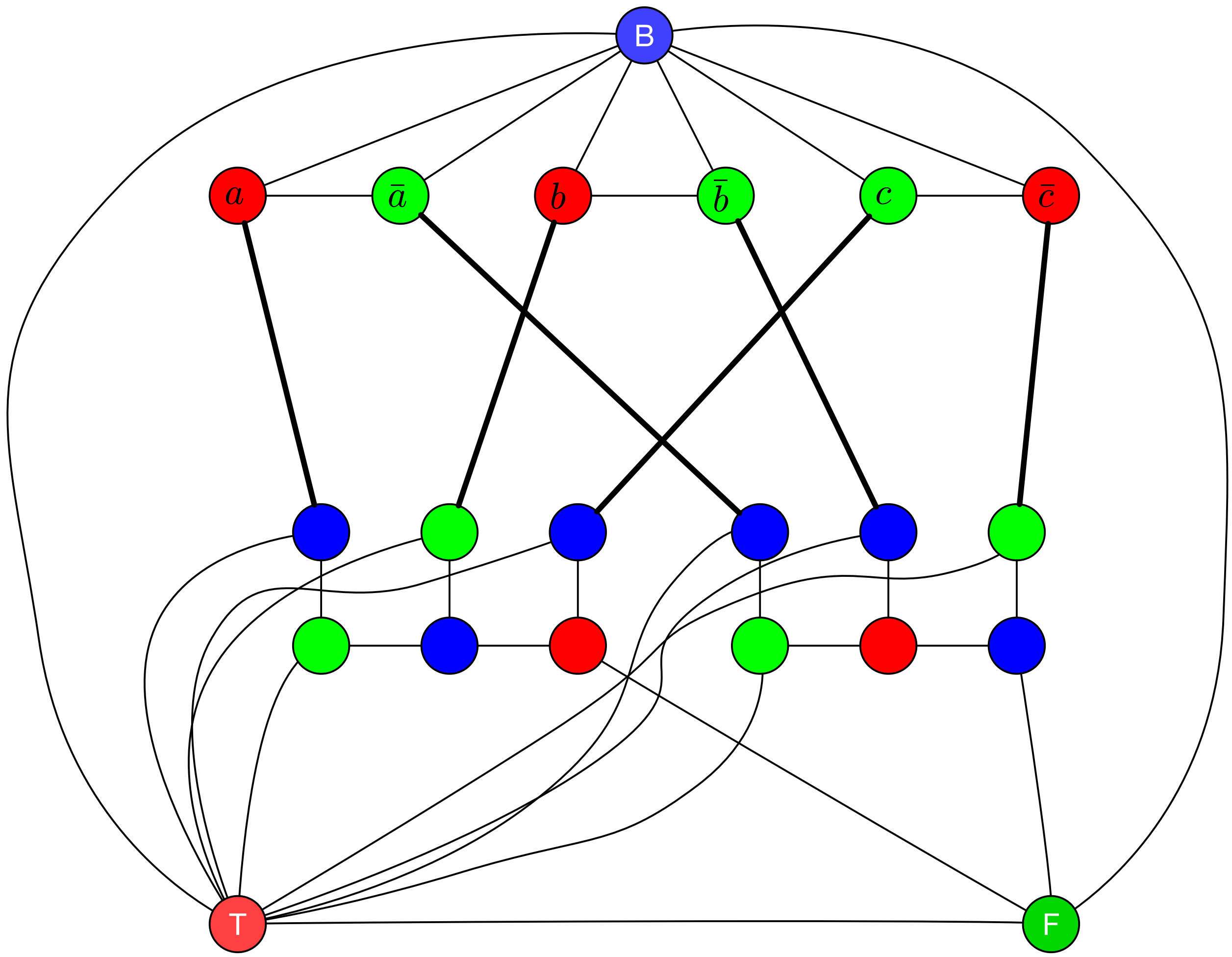
On the right, we have a white node connected to a blue node and the F-node, that has to be red.



$$a \vee b \vee c \qquad \bar{a} \vee \bar{b} \vee \bar{c}$$

The other nodes need to be filled in. The second row cannot be red and has to be filled in with green and blue. This leaves only a few cases, and we can come up with a coloring that fulfills the rules.

So, this is our final coloring, which is not uniquely determined.



$$a \vee b \vee c \qquad \bar{a} \vee \bar{b} \vee \bar{c}$$

The point of coloring is not to find a way to satisfy a 3-literal conjunctive normal form. Our exponential algorithm of filling in a truth table is so much better. It is to illustrate that if we can determine whether a graph is three-colorable, then we can determine whether the corresponding 3-SAT problem is solvable. The graph is larger than the 3-SAT problem, but it has 3 + 2*(number of variables) + 6*(number of clauses) nodes and a similar linear number of edges. Thus, if I have a 3-SAT problem and if I can solve 3-colorable in poly-time, then I can construct the graph of the 3-SAT problem with a linear blow-up, use the poly-time 3-colorable algorithm, and then deduce the solution of the 3-SAT problem from the coloring. I read the solution in the first row, where the coloring of the literals tells me which assignment makes the formula true. In our case, a and b are red and so is $\neg c$.