CS 325 GA#3
Team Members:
Noelcke, Thomas
Turkington, Kevin
Spencer, Jonah
Struth, Joseph

Lights And Switches

Pseudo Code:

The pseudo code for our solution to the Lights and Switches problem is shown below:

1. First we read in the state of the lights and switches from the file.
2. We then loop through every light in the Building Converting the state of the light and switches into 2 CNF (Conjunctive Normal Form).
    a. We can do this using the following where x1 and x2 represent the two switches attached to each light.
        i. If the light is on then we know (x1 v x2) ^ (!x1 v !x2)
        ii. If the light is off then we know (x1 v !x2) ^ (!x1 v x2)
    b. Doing this results in a long CNF statement.
3. We then take our 2SAT solver and see if this statement is satisfiable.
    a. If yes then we print yes to the file.
    b. If no then we print no to the file.

Time Complexity:

N = the number of Lights

The complexity of the above code is O(N * Number of switches connected at each N). It is given that there are no more than 2 switches attached to each light bulb, so the complexity of this is: O(N)

Proof of Correctness:

To prove this algorithm we must prove that our reduction correctly represents the state of the lights and switches correctly. To do this let x1 represent the state of switch one let x2 represent the state of switch 2 and let l1 represent the state of the light connected to these two switches. Also note that = represents triple bar or iff.

Generally we can say that in order for the light to be one one switch must be in the off position and one switch must be in the off position or: x1 XOR x2 = l1.

Or ((x1 or x2) and (!x1 or !x2)) = I1
From this we can also say this !(x1 XOR x2) = ! I1.
OR ((!x1 or x2) and (x1 or x2)) = !I1.
This statement essentially means that in order for the light to be off both switches must be on or both must be off. So we can say:

(((x1 or x2) and (!x1 or !x2)) = I1) = (((!x1 or x1) and (x1 or !x2)) = !I1)))

This must hold true for all possible inputs if we are to say that this accurately represents the state of one light.

By truth table we can show that this is true:

x1	x2	I1	(((x1 or x2) and (!x1 or !x2))	=	f1)	=	(((x1 or !x2) and (!x1 or x2))	=	!f1)
t	t	t	f	f	t	t	t	f	f
t	t	f	f	t	f	t	t	t	t
t	f	t	t	t	t	t	f	t	f
t	f	f	t	f	f	t	f	f	t
f	t	t	t	t	t	t	f	t	f
f	t	f	t	f	f	t	t	f	f
f	f	t	f	f	t	t	t	f	f
f	f	f	f	t	f	t	t	t	t

Because this holds true for all possible configurations of the system we can use this formula to say that:

if I1 then ((x1 or x2) and (!x1 or !x2))
If !I1 then ((!x1 or x2) and (x1 or !x2))

We can use this to form a CNF statement that represents the possible state of the switches of the lights and switches system. If this statement is satisfiable than we can say it is possible to turn off all the lights. If this system is not satisfiable than we can say that there isn't a state of the system where all the lights are off.