Default Report (Copy)

LTL colleagues summer 2021

August 18, 2021 6:55 PM MDT

Q2 - Timing on 1st page (0 questions)

Times in seconds

#	Field	Minimum	Maximum	Mean	Std Deviation	Count
1	Timing - First Click	0	179	12	33	29
2	Timing - Last Click	0	186	20	39	29
3	Timing - Page Submit	11	3739	180	674	29
4	Timing - Click Count	0	18	3	5	29

Q20 - Timing on 2nd page (4 trace questions)

Times in seconds

#	Field	Minimum	Maximum	Mean	Std Deviation	Count
1	Timing - First Click	0	71	28	17	29
2	Timing - Last Click	86	414	160	75	29
3	Timing - Page Submit	87	423	166	75	29
4	Timing - Click Count	6	41	11	8	29

Q32 - Timing on 3rd page (3 LTL to English questions)

Times in seconds

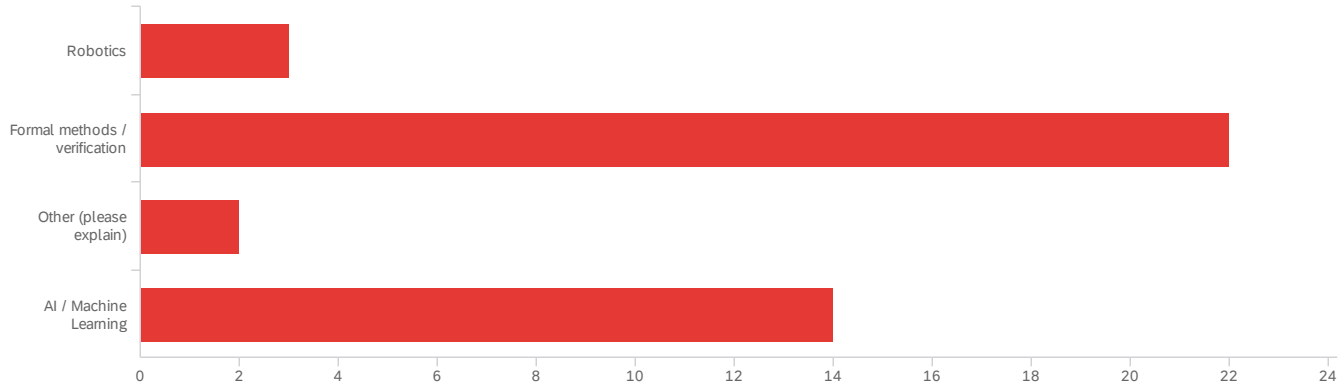
#	Field	Minimum	Maximum	Mean	Std Deviation	Count
1	Timing - First Click	2	36	19	9	29
2	Timing - Last Click	84	5254	544	982	29
3	Timing - Page Submit	105	5255	569	977	29
4	Timing - Click Count	4	91	21	19	29

Q42 - Timing on 4th page (2 English to LTL questions)

Times in seconds

#	Field	Minimum	Maximum	Mean	Std Deviation	Count
1	Timing - First Click	1	610	34	109	29
2	Timing - Last Click	47	686	211	170	29
3	Timing - Page Submit	71	700	244	166	29
4	Timing - Click Count	3	73	12	14	29

Q44 - Which areas do you identify with? Select all that apply.



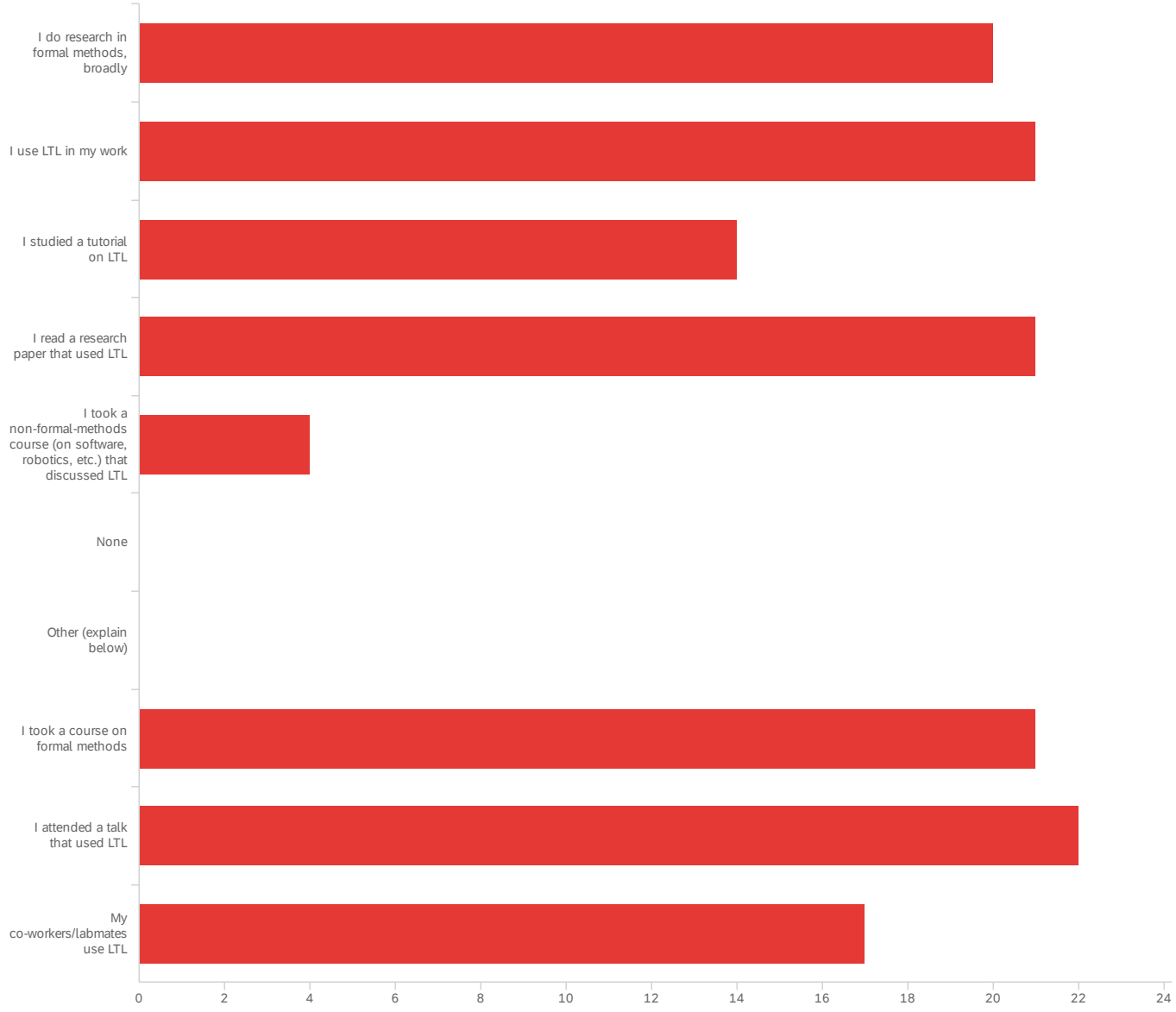
Q44_14_TEXT - Other (please explain)

Other (please explain)
Programming languages / software engineering
HCI

Q47 - If you have any comments about your answers, or about this survey, please let us know.

If you have any comments about your answers, or about this survey, please l...
very hard to translate into English, since it seems to depend so much on the context, and the panel is a bit of an unfamiliar one.
None
The English to LTL questions are tricky to get right.
I see now that the until is labeled as strong, but I forget what that means. :-(
Will I get credits? :-)
Did you consider making the translation-to-English question multiple choice? You could have asked more questions then it seems.
I'm not sure about my answers... Never did this kind of exercise.
This was actually fun :)

Q43 - What is your prior exposure to LTL? Select all that apply.

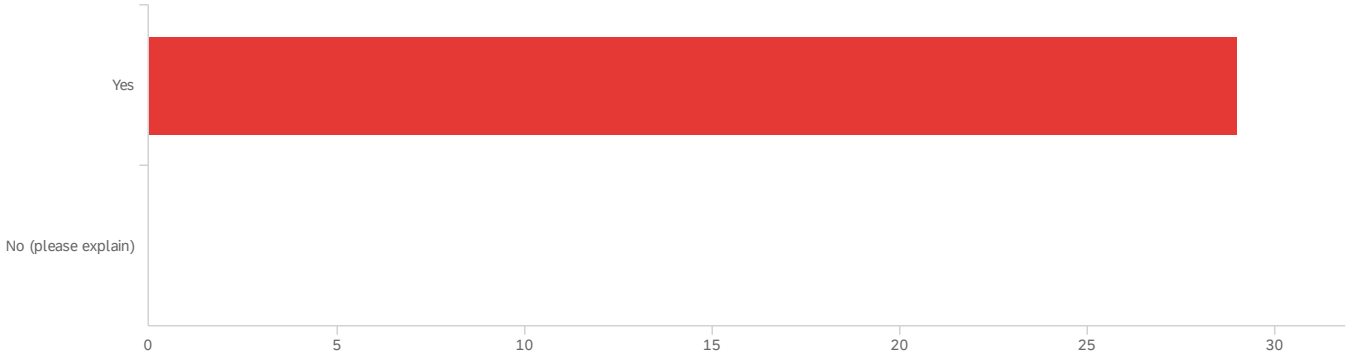


Q43_7_TEXT - Other (explain below)

Other (explain below)

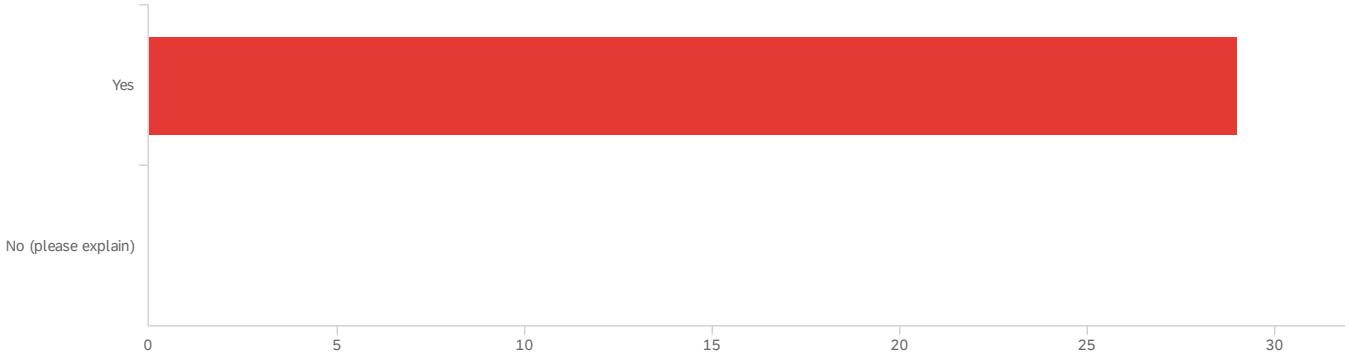
Q7 - Does the example make sense to you?*

expecting: Yes



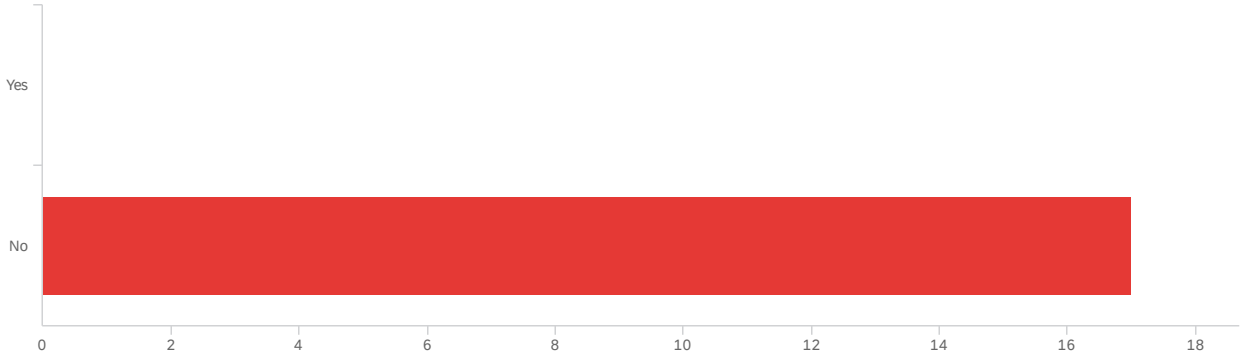
Q9 - Does the example make sense to you?*

expecting: Yes



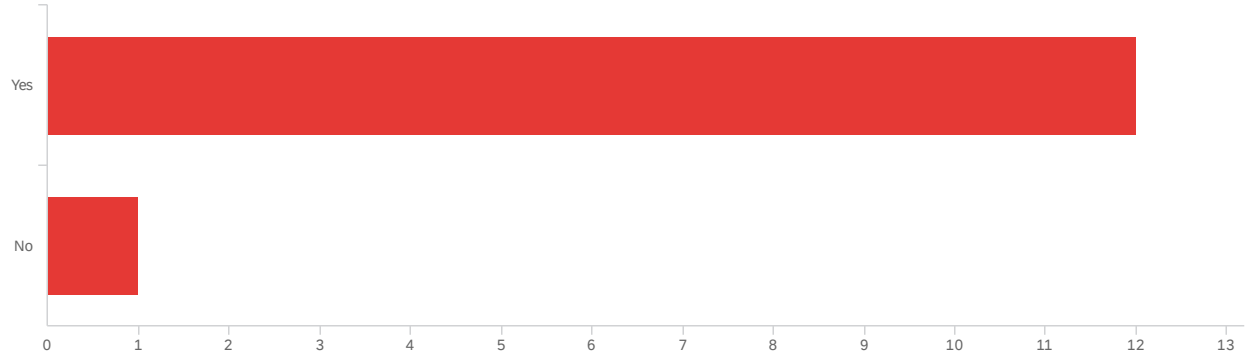
Q11 - Q. Is the formula Red satisfied by this trace?*

correct = No, trace = {GB} {RGB}*



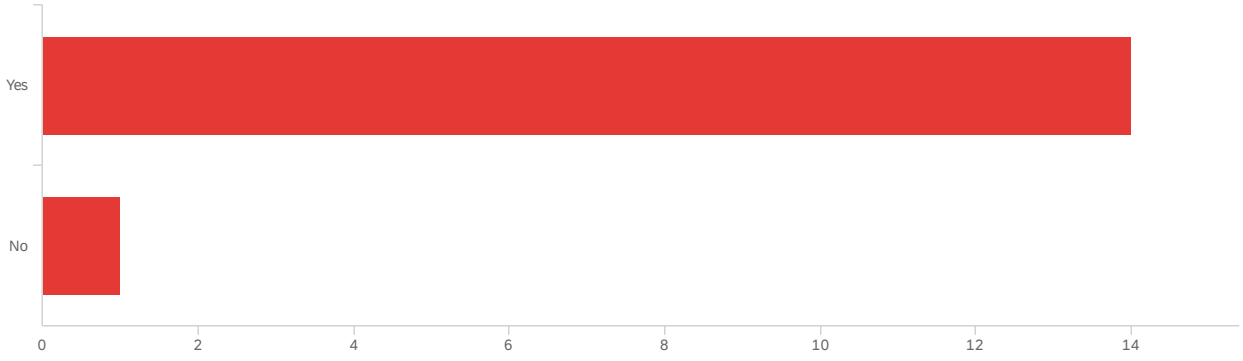
Q13 - Q. Is the formula always (Red implies after (after (after (Red)))) satisfied by this trace?*

correct = Yes, trace = {} {RGB} {RGB} {} {RGB}*



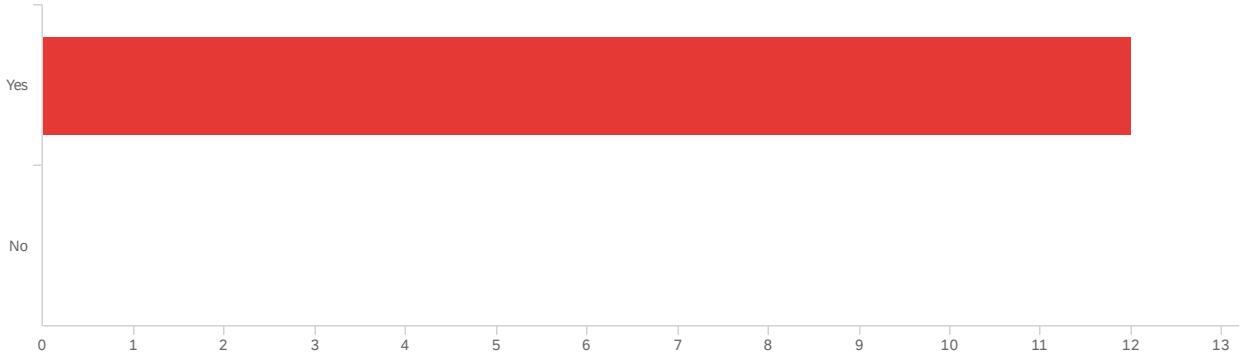
Q14 - Q. Is the formula (after (Red)) until (after (Green)) satisfied by this trace?*

correct = Yes, trace = {RB} {RB} {RB} {RGB} {B}*



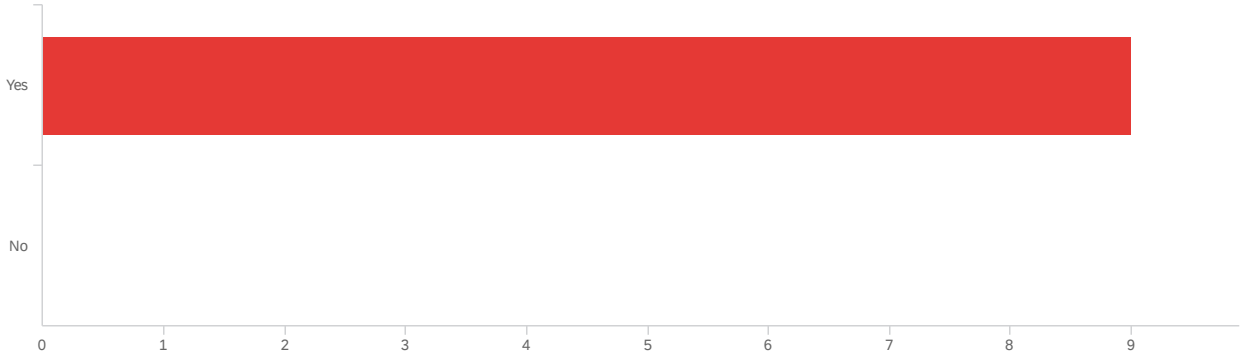
Q15 - Q. Is the formula { eventually (Red) } and { eventually (Green) } satisfied by this trace?*

correct = Yes, trace = {} {G} {} {} {R}*



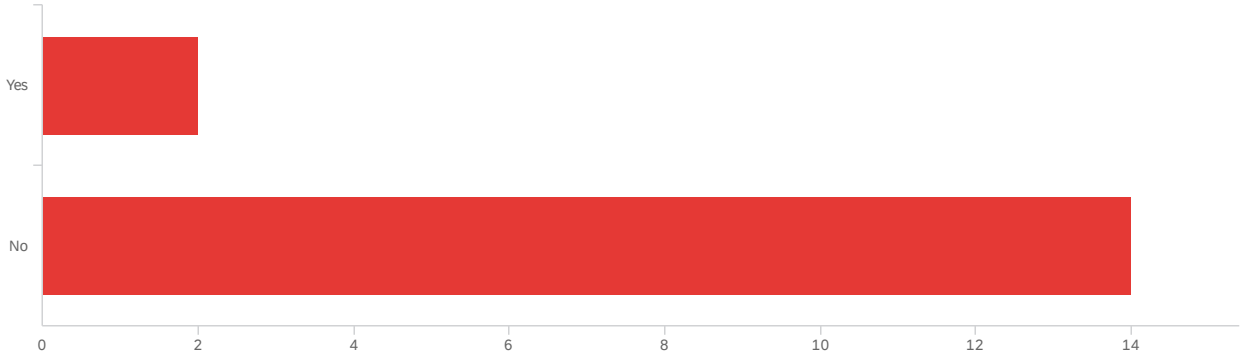
Q16 - Q. Is the formula after (after (eventually (Red))) satisfied by this trace?*

correct = Yes, trace = {RGB} {RGB} {RGB} {RGB} {RGB}*



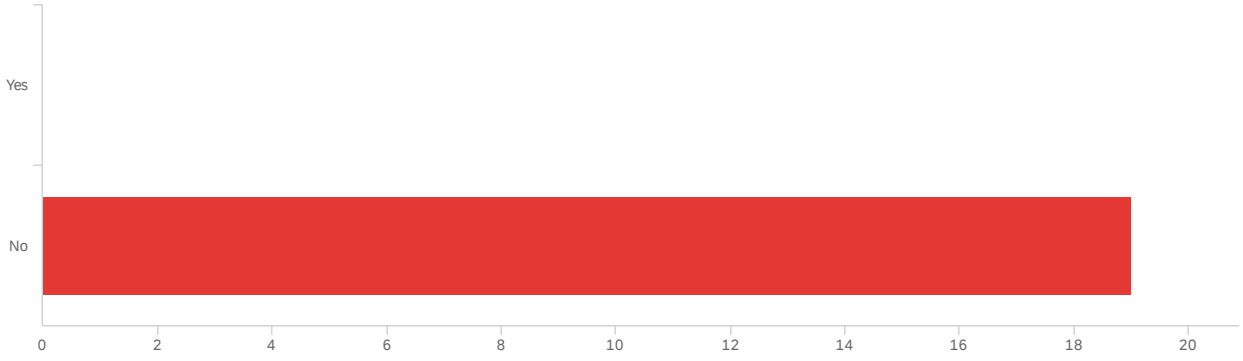
Q17 - Q. Is the formula (Red) until (Blue) satisfied by this trace?*

correct = No, trace = {R} {R} {R} {R} {R}*



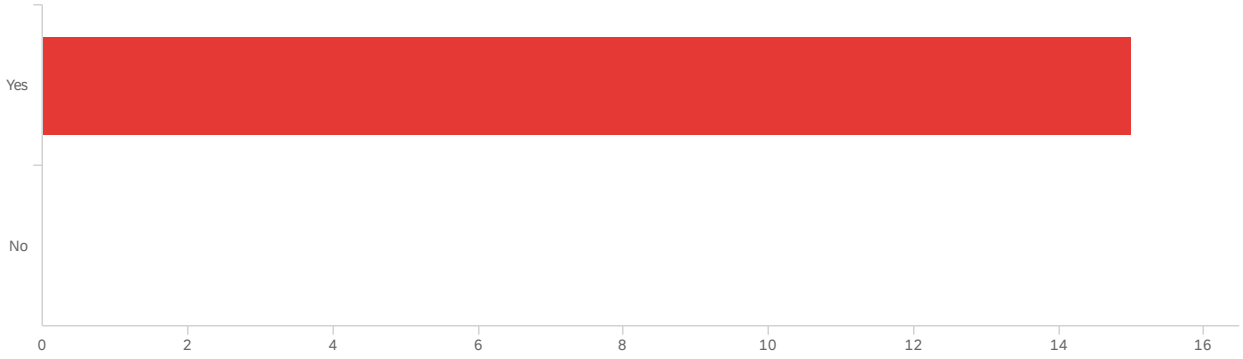
Q18 - Q. Is the formula eventually (always (Red)) satisfied by this trace?*

correct = No, trace = {} {RGB} {} {RGB} {}



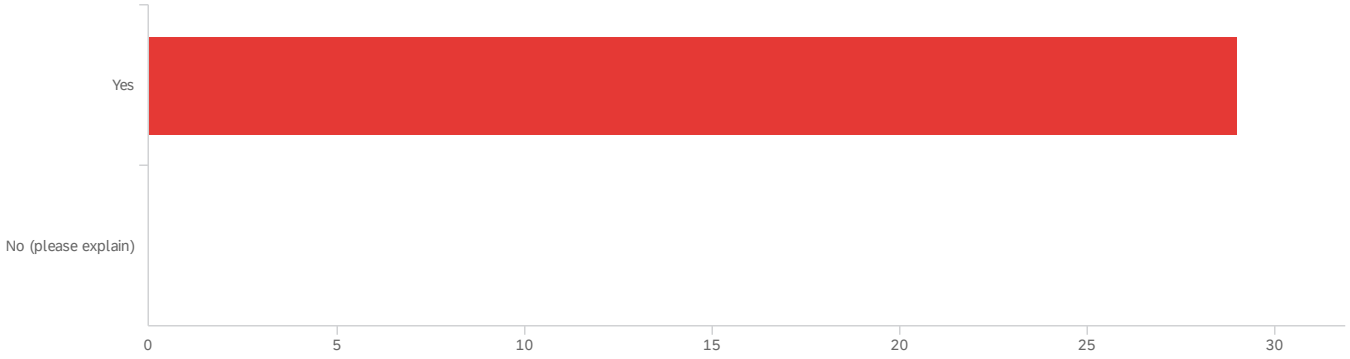
Q19 - Q. Is the formula always (Red implies Green) satisfied by this trace?*

correct = Yes, trace = {} {} {} {} {}*



Q23 - Does the example make sense to you?*

expected = Yes

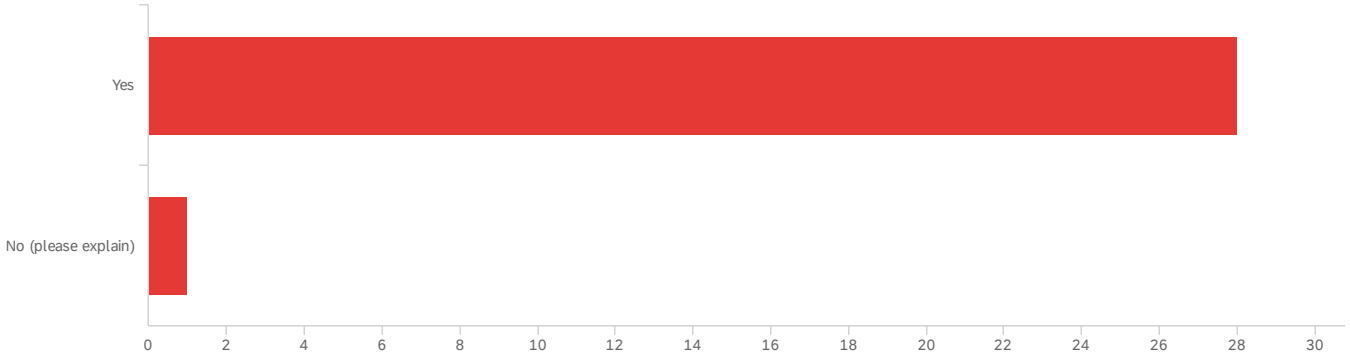


Q23_3_TEXT - No (please explain)

No (please explain)

Q25 - Does the example make sense to you?*

expected = Yes



Q25_3_TEXT - No (please explain)

No (please explain)
"then turns off" seems like a permanent thing. Is it?

Q27 - Red implies after (after (after (Red)))

correct = "If Red initially, then Red again in 4th state."

Red implies after (after (after (Red)))
If Red is on at some timestep, then it must be on at 3 timesteps after that timestep.
If the Red light is on for the first time step, it's also on exactly three steps after.
if the panel starts out with red lit, it will be lit three steps later
when the red light is on, it keeps being red for the following 3 next states.
The red light being on implies that three states later it will be on as well.
Red is on for 0 or more states; the 3rd state (next-next-next state) after each state where red is on will have red be on.
If the red light is on in a given state, the red light will be on again in the third following state
If the first state has red light on, the fourth state will also have red light on.
If Red is on initially it will be on also in the third state after the first.
If red is on in the beginning, it needs to also be on at step 4.
If the Red light is on then in three steps will be on again.
if the red light is on in three steps will be on again.
If the red light is on in the first state, it is on in the fourth state
If Red is true now then it is also true after three steps
If the red light is initially on, it will be also at the fourth step
If the Red light is on now, then it is on also in 3 timesteps.
If the Red light is on now, then it will be on in the third step from now.
If the Red light is on in the first state, then the Red light is on in the fourth state.
If the red light is on in the first state, it is also on in the fourth state.

Q28 - after (after (eventually (after (Red))))

correct = "Red at 4th state or later"

after (after (eventually (after (Red))))

Red will be one at some point from the 4th state onwards

Red is on in the 4th state or later.

ignoring the first two states, the panel shows red at some point

The Red light turns on at least once from the third state.

Starting at the second state, there will eventually be a state followed by a state in which the red light is on.

red is on in some state after the first two states

The red light must turn on sometime after the 3rd state.

After two time steps, at some point in the future there is a state where the red light turns on in the time step immediately following it.

Red will be eventually on but not before the forth state of the execution.

The red light is on at some point after the third hop.

In three steps the light will eventually be on.

in the next 2 steps, red will eventually turn on after one step

Red is on after three or more steps

The red light is on at least in one timestep from the third onward.

Counting from state 3, the Red light is eventually on

After two time steps, there will eventually be a state when the next state will have the red light on.

The red light is on in some state after the third (regardless of whether it was on in one of the first three states).

Q29 - { eventually (Red) } implies { always (Blue) }

correct = "If ever Red, then Blue always from first state"

{ eventually (Red) } implies { always (Blue) }

If Red is on at any state in the trace, then Blue is on at all states in the trace.

Either the Red light is always off, or the Blue light is always on.

if the panel can ever show red, then it must always show blue

Red is on at some point, after which blue is always on

If Red light is on at some state, Blue should always be on

If the red light is on at some state in the future, the blue light must be on for every state.

If ever the red light turns on, the blue light also turns on and stays on forever.

If Red ever turns on, it darn well better be the case that Blue starts out on and never goes off!

If there is a state where the blue light is off, then the red light is always off.

if eventually the Red light is on then the Blue light is never off.

If eventually the red light is on then the blue light will never get off.

Either Red is never on, or Blue is always on.

if red is on at some point, then blue will be always on

If the red light is on in some state, the blue light is on in all states

If Red is ever true in the future then blue holds forever

If the red light will be eventually on, then the blue light is always on from the beginning.

If I see Red on at some point, then I always see Blue on

If eventually the red light is going to be on, then the blue light has to remain on all the time.

If the Red light turns on at some point, it means that the Blue light is always on from the very beginning.

Q30 - ((Red) until (Blue)) and always (Red)

correct = "Red forever, Blue eventually"

((Red) until (Blue)) and always (Red)

Red is always on.

The Red light is always on and eventually the Blue light is on.

The Red light is always on and the Blue light eventually turns on.

The statement can be never satisfied.

The red light is on always, and the blue light turns on after 0 or more states

The red light is on for every state, and the blue light also turns on at some state.

Red light is always on and Blue light is on at some state

The blue light must turn on at some point and the red light must always be on.

The red light is always on and there is always a future state where the blue light will turn on (since it is strong 'until').

Red must always be on. (I am not sure which flavor of "until" that is. If it's the kind that insists that the part after the "until" happens, then I'd add "and Blue must eventually switch on.")

Eventually the blue light turns on (once, then may turn off again), while red light remains always on.

Red is always on, and Blue comes on a some point.

The red light is always on and the blue light is on in some state

Red is always on and Blue is eventually on

The Red light is on for zero or more states until a state in which the Blue light is on, and in addition to all that, the Red light is always on

The Red light is always on. The Blue light is on in the second state.

The red light is on in all states and the blue light is on in at least one state.

Q31 - always (Red implies (after (not Red) and after (after (Red))))

correct = "If Red then blinks forever"

always (Red implies (after (not Red) and after (...

If red is on at some timestep, then it must be off at the next timestep and on at the timestep after that.

Whenever the Red light is on, it starts alternating between on and off at each state.

whenever red is on, the next state will not be red and the next two state will be red.

Whenever the red light is on, it is off in the next state and then on again the state after that.

Whenever the red light is on, the red light is off for the first following state and back on the second following state.

Every time the red light is on, it is followed by a state with red off and then a state with red on.

If at any state Red light is on, Red light should not be on in the state after that and should be on again in the state after that.

Red continuously toggles between on and off at each state transitions.

From the time the red light goes on, it stays blinking.

Whenever Red goes on, it goes off in the next step, then on again in the next next step. That is, Red alternates on and off

whenever red is on, in the next timestep red is off and in the next two steps red is on again

Whenever the Red light is on, it turns off in the next timestep, and turns on again after that.

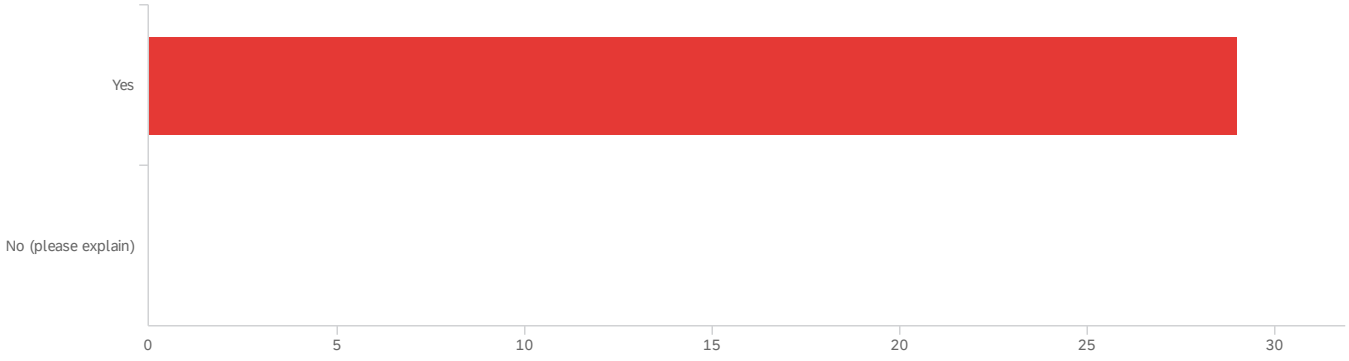
If Red is on at some point, then from that point on it starts alternating between on and off forever.

Whenever the red light is on it goes off in the next time step and then turns on again in the following time step.

For every state S, if the Red light is on in S, then the Red light is off in the next state (after S) and on in the second-next state (after S).

Q35 - Does the example make sense to you?*

expected = Yes



Q37 - Whenever the Red light is on, it is off in the next state and on again in the state after that.

correct = G(Red => X(!Red) and XX(Red))

Whenever the Red light is on, it is off in the next state and on again in t...
always (Red implies (after not Red) and (after after Red))
always (red implies (after (not red) and after after red))
always (Red implies (next not Red and next next Red))
always (Red implies (after (not Red) and after (after(Red))))
Red implies ((after not Red) and (after after Red))
always (Red implies ((next not Red) and (next next Red)))
Always (Red implies (After !Red and After After Red))
always (Red implies (after (not Red and after Red)))
... the last formula from the previous page ...
(eventually (red)) implies (after (not red and (after (red)))
always(Red -> (after((not Red) and after (Red))))
G(Red -> (X(~Red) AND XX(Red)))
always (Red implies ((not after Red) and (after after Red)))

Q38 - The Red light is on in exactly one state, but not necessarily the first state.

correct = { !Red } U { Red and X(G(!Red)) } ... OR ... F(Red) and G(Red => X(G(!Red)))

The Red light is on in exactly one state, but not necessarily the first sta...
~Red U (Red /\XG(~Red))
F (a & X G!a)
(not Red) until (Red and (next always (not Red)))
eventually (Red and after (always (not Red)))
(eventually Red) and always (Red implies after always not Red)
Eventually (Red) and (Red implies after (always not Red))
(not Red) U (Red and (G(not red)))
Eventually (Red & After (Always (! Red)))
(not Red) until (Red and after always not Red)
(eventually Red) and (always (Red implies after (always not Red)))
(eventually Red) and (Red implies (after(always (not Red))))

Q39 - The Red light cannot stay on for three states in a row.

correct = G(Red and X(Red) => XX(!Red)) ... OR ... !F(Red and X(Red) and XX(Red)) ... ETC

The Red light cannot stay on for three states in a row.
Red implies (after(not red) or after(after(not red)) or after(after(after(not red))))
G ((a & X a) -> (!XXa))
always not (Red and after Red and after after Red)
always ((not Red) or (after not Red) or (after after not Red))
always not (red implies (after red and after after red))
G(Red & X (Red) implies XX(not red))
always(not(Red and after(Red and after(Red))))
G((Red and X(Red)) => XX(not Red))
G(~(Red and X(Red) and XX(Red))

Q40 - Whenever the Red light is on, the Blue light will be on then or at some point in the future.

correct = G(Red => F(Blue))

Whenever the Red light is on, the Blue light will be on then or at some poi...
always(Red -> eventually Blue)
always (Red implies eventually Blue)
always (Red implies eventually Blue)
always (Red implies (eventually Blue))
always (Red implies (eventually Blue))
always (Red implies eventually Blue)
always (Red implies (eventually Blue))
always (Red implies (eventually (Blue)))
Red implies (eventually Blue)
Red implies (eventually (Blue))
G(Red implies F(Blue))
G(Red => F(Blue))

Q41 - The Red light is on for zero or more states, and then turns off and remains off in the future.

correct = Red U G(!Red) ... OR ... F(!Red) and G(!Red => X(G(!Red)))

The Red light is on for zero or more states, and then turns off and remains...
Red until (eventually (always (not red))
(Red) until (always (not Red))
Red implies(after(Red) or (always not(Red)))
Red until (not Red) and (not Red implies after not Red)
(Red) until (always (not Red))
(red) until (always (not red))
eventually (not Red and always not Red)
Red until (Always !Red)
Red Until (always not Red)
Red until (always not Red)
(red) until (always (not red))
(Red) Until (Always (! Red))
Red until (always not Red)

End of Report