COL:750

Foundations of Automatic Verification

Instructor: Priyanka Golia

Course Webpage

https://priyanka-golia.github.io/teaching/COL-750/index.html

LTL: Semantics

We interpret our temporal formulae in a discrete, linear model of time.

$M = < N, I >$, where N is a set of Natural number and $I : N \mapsto 2^{\Sigma}$

I maps each Natural number (representing a moment in time) to a set of propositions

Let $\pi = a_0, a_1, a_2, \ldots$ $\pi(i) = a_i$ AP at i^{th} level.

$\pi^i = a_i, a_{i+1}, a_{i+2}, \ldots$ Suffix of π

LTL: Semantics

Semantics with respect to a given Trace (or Path) π

Let $\pi = a_0, a_1, a_2, \ldots$ $\quad \pi(i) = a_i$ AP at i^{th} level. $\quad \pi^i = a_i, a_{i+1}, a_{i+2}, \ldots$ Suffix of π

$\pi \models p$ Iff $p \in \pi(0)$ $\qquad \pi^i \models p$ Iff $p \in \pi(i)$

$\pi \models \mathbf{N}\ F_1$ Iff $\pi^1 \models F_1$ $\qquad \pi^i \models \mathbf{N}\ F$ Iff $\pi^{i+1} \models F_1$

$\pi \models F_1\ \mathbf{U}\ F_2$ Iff $\exists j \geq 0,\ \pi^j \models F_2$, and $\pi^i \models F_1$ for all $0 \leq i < j$

$\pi \models \Diamond\ F_1$ Iff $\exists j \geq 0,\ \pi^j \models F_1$

$\pi \models \Box\ F_1$ Iff $\forall j \geq 0,\ \pi^j \models F_1$

$\pi \models \Box \Diamond\ F_1$ Iff $\exists^\infty j \geq 0,\ \pi^j \models F_1$ $\qquad \exists^\infty = \forall i \geq 0, \exists j \geq i$

$\pi \models \Diamond \Box\ F_1$ Iff $\forall^\infty j \geq 0,\ \pi^j \models F_1$ $\qquad \forall^\infty = \exists i \geq 0, \forall j \geq i$

LTL: Semantics Kripke Structure

AP — is a set of atomic propositions (Boolean valued variables, predicates)

Kripke structure over AP as a 4-tuple $M = (S, I, R, L)$

S = a finite set of states.

I = a set of initial states $I \subseteq S$

R = a transition relation $R \subseteq S \times S$

L = a labelling function $L : S \rightarrow 2^{AP}$

LTL: Semantics Kripke Structure

Kripke structure over AP as a 4-tuple $M = (S, I, R, L)$

S = a finite set of states. $S = \{s_1, s_2, s_3\}$

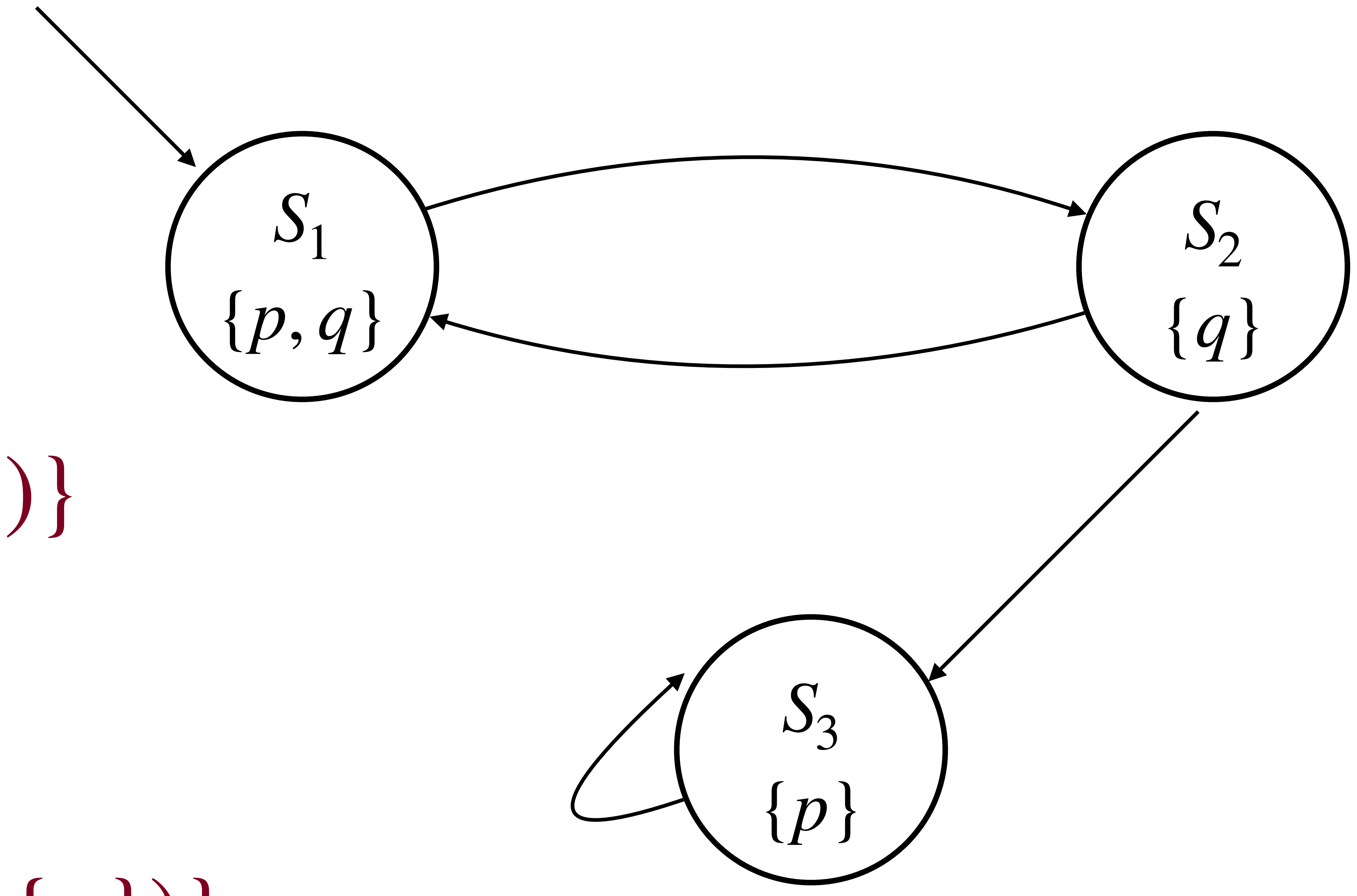
I = a set of initial states $I \subseteq S$ $I = \{s_1\}$

R = a transition relation $R \subseteq S \times S$

$$R = \{(s_1, s_2), (s_2, s_1), (s_2, s_3), (s_3, s_3)\}$$

L = a labelling function $L : S \rightarrow 2^{AP}$

$$L = \{(s_1, \{p, q\}), (s_2, \{q\}), (s_3, \{p\})\}$$



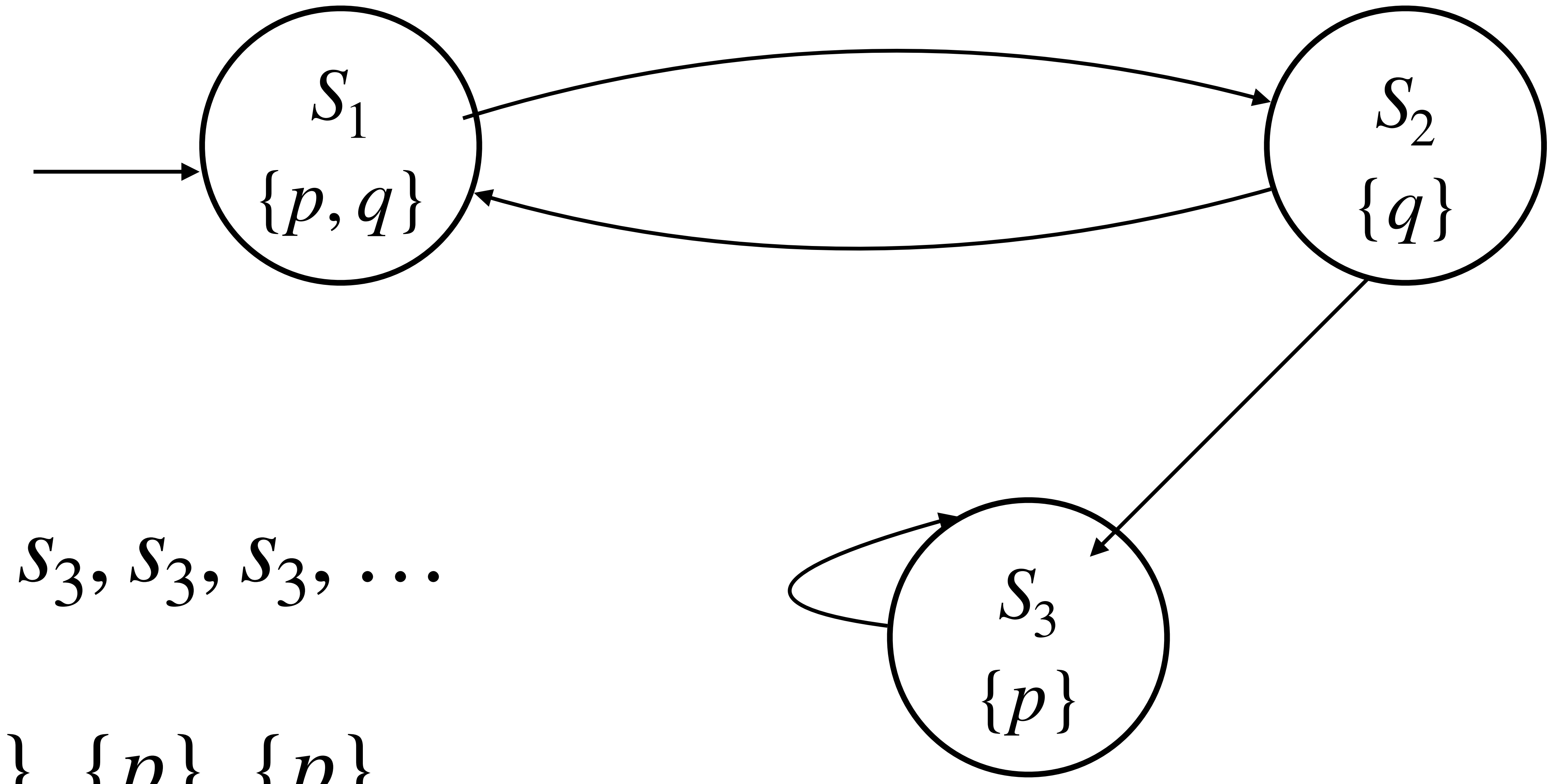
AP = $\{p, q\}$

LTL: Semantics Kripke Structure

Kripke structure over AP as a 4-tuple $M = (S, I, R, L)$ AP = $\{p, q\}$

$S = \{s_1, s_2, s_3\}$ $I = \{s_1\}$ $R = \{(s_1, s_2), (s_2, s_1), (s_2, s_3), (s_3, s_3)\}$

$L = \{(s_1, \{p, q\}), (s_2, \{q\}), (s_3, \{p\})\}$



M may produce a path $w = s_1, s_2, s_1, s_2, s_3, s_3, s_3, s_3, \ldots$

π^{s_1} $\pi = \{p, q\}, \{q\}, \{p, q\}, \{q\}, \{p\}, \{p\}, \{p\}, \ldots$

LTL: Semantics

Kripke Structure

Given a kripke structure M and a path π in M, a state $s \in S$, and an LTL formula F:

1. $< M, \pi > \models F$ iff $\pi^{s_o} \models F$, where s_o is initial state of π
2. $< M, s_o > \models F$ iff $< M, \pi > \models F$ for all paths starting at s_o.
3. $< M > \models F$. iff $< M, s_o > \models F$ for every $s_o \in I$, where I initial states of M.

LTL: Semantics

A formula F is satisfiable if there exists at least one Kripke Structure M, and at least one initial state s_o such that:

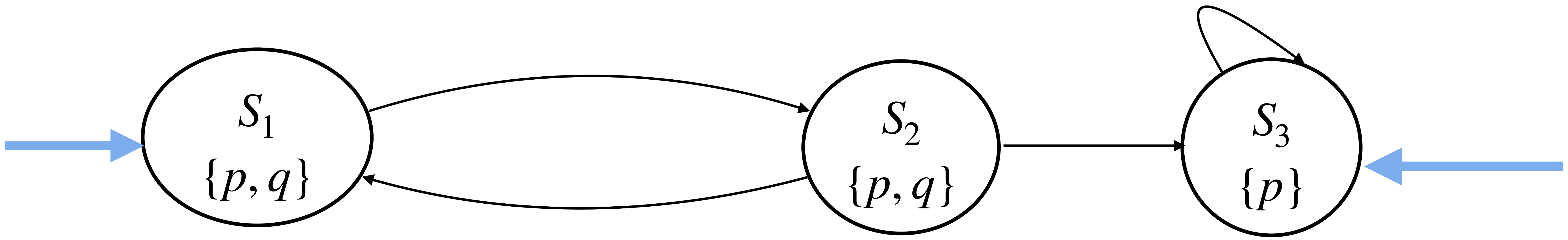
$$< M, s_o > \models F$$

A formula F is valid if for all Kripke Structures M, and for all initial states s_o:

$$< M, s_o > \models F$$

LTL model checking — Given formula F, and Kripke Structure M checks if $< M, s_o > \models F$ holds for every initial state $s_o \in I$

LTL: Semantics



Does $M \models \Box p$? Yes, $< M, s_1 > \models \Box p$ and $< M, s_3 > \models \Box p$

$\pi_1^{S_1} = < \{p,q\}\{p,q\}, \{p,q\}, \{p,q\} \ldots >$ $\pi_2^{S_1} = < \{p,q\}\{p,q\}, \{p,q\}, \{p,q\}, \{p\}, \{p\} \ldots >$ $\pi_3^{S_3} = < \{p\}, \{p\} \ldots. >$

Does $M \models \mathbf{N}(p \wedge q)$? No, $< M, s_1 > \models \mathbf{N}(p \wedge q)$, but $< M, s_3 > \not\models \mathbf{N}(p \wedge q)$

Does $M \models \Box(\neg q \rightarrow \Box(p \wedge \neg q))$? Yes

Does $M \models q\ \mathbf{U}(p \wedge \neg q)$? No, $< M, \pi_1 > \not\models q\ \mathbf{U}\ (p \wedge \neg q)$

LTL implicitly quantifies "universally" over paths —

$< M, s_o > \vDash F$ iff $< M, \pi > \vDash F$ for all paths starting at s_o.

$F = \Diamond(p)$ F is True if for all the paths, eventually p is True.

Does there exists a path where eventually p is True?

Is it possible to get to a state where the machine is not ready but it started ?

$\Box \neg(\neg ready \wedge started)$

One way to do is: $\Box \neg(p)$

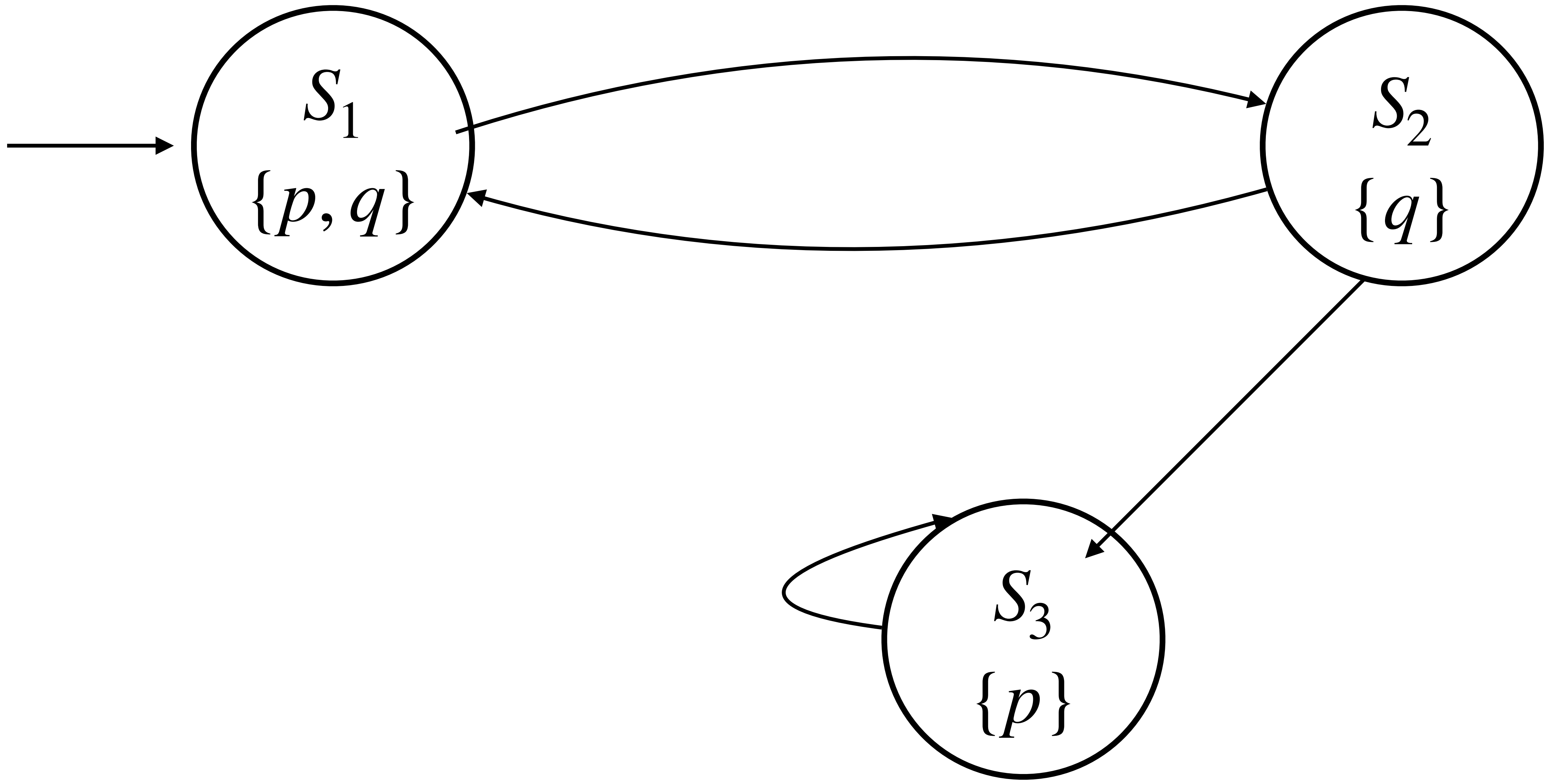
But how to model:
There exists a path where, from some state onward, all future states avoid deadlock?

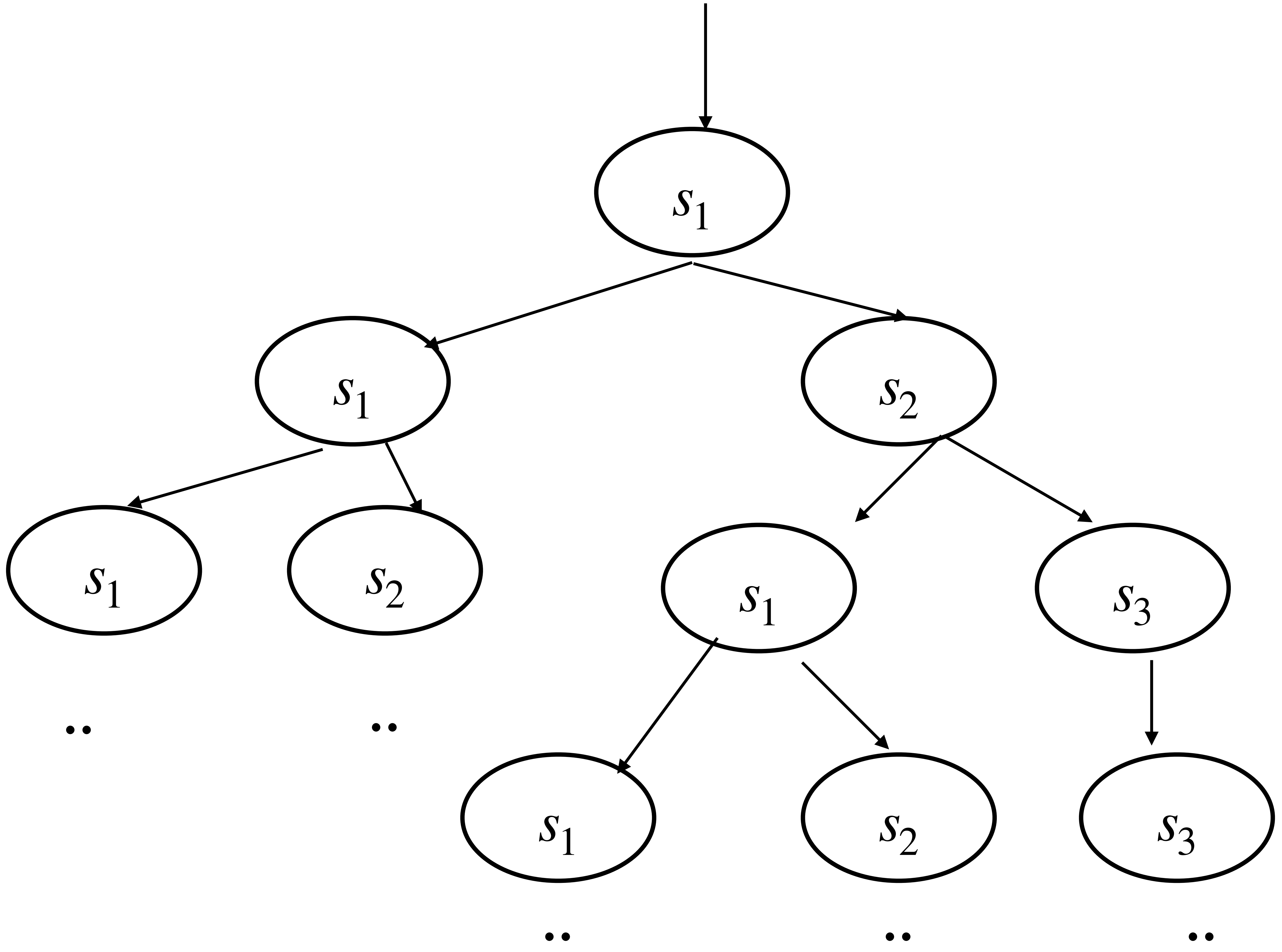
We need path quantifiers!!!

Computation Tree Logic (CTL)

LTL — deals with paths or traces.

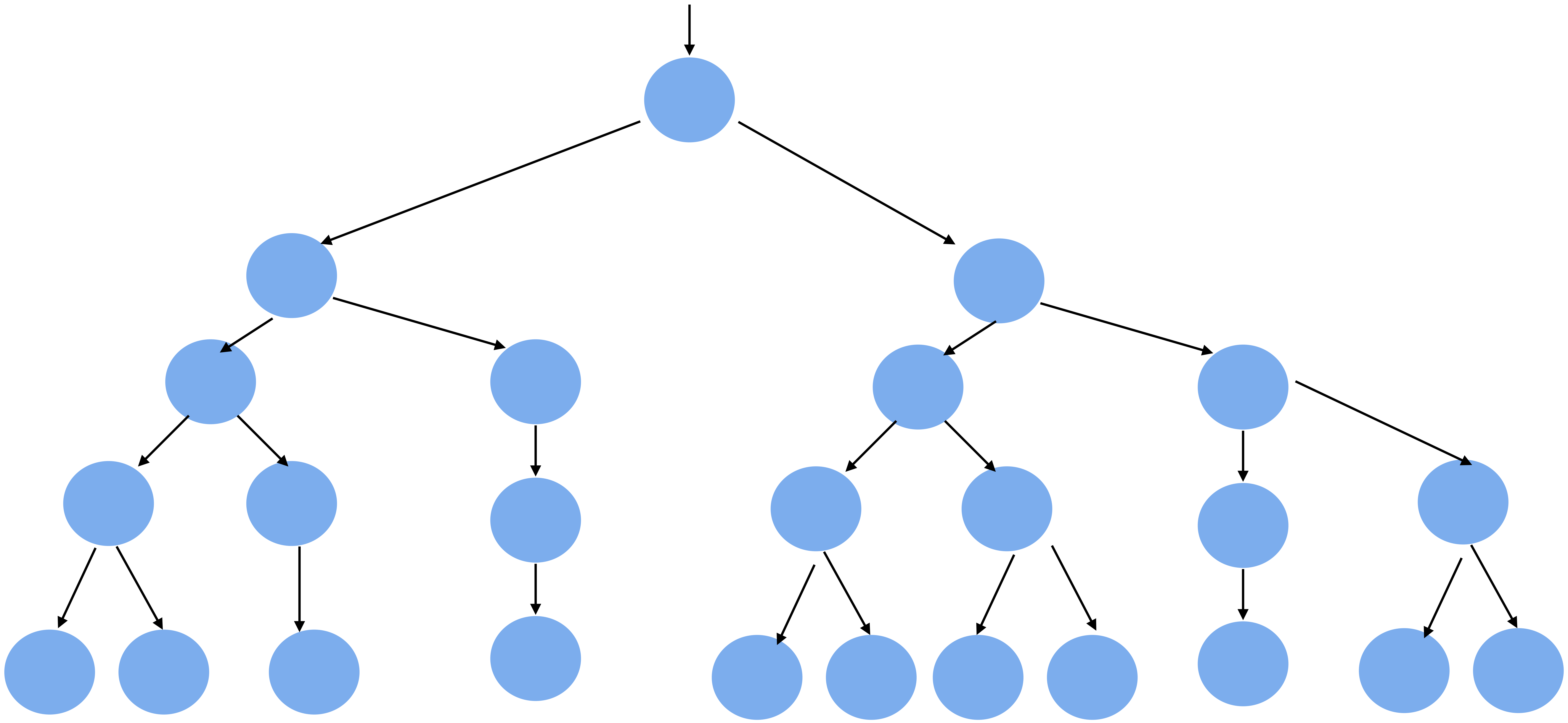
CTL — branching time structure (Trees)





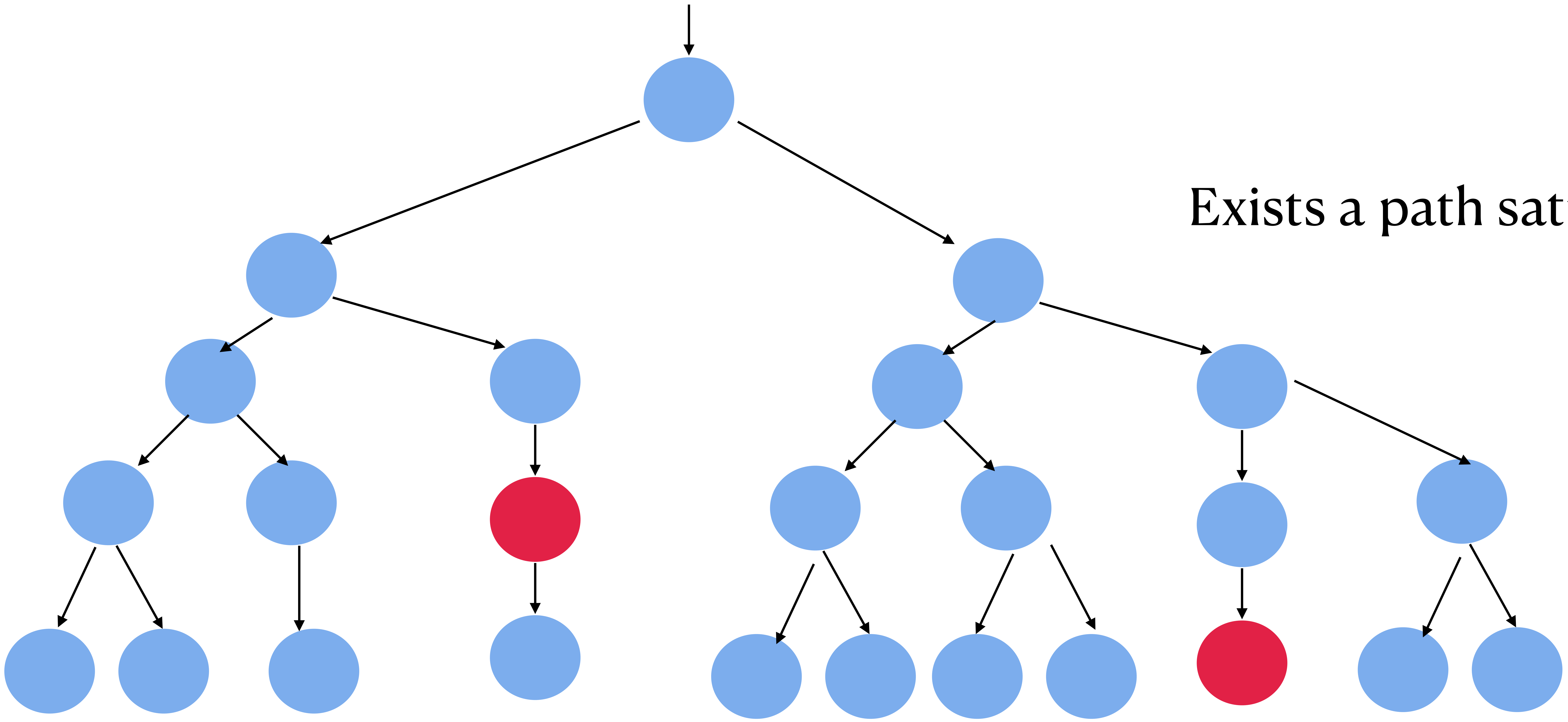
Computation Tree Logic (CTL)

Talks about properties of trees!



Computation Tree Logic (CTL)

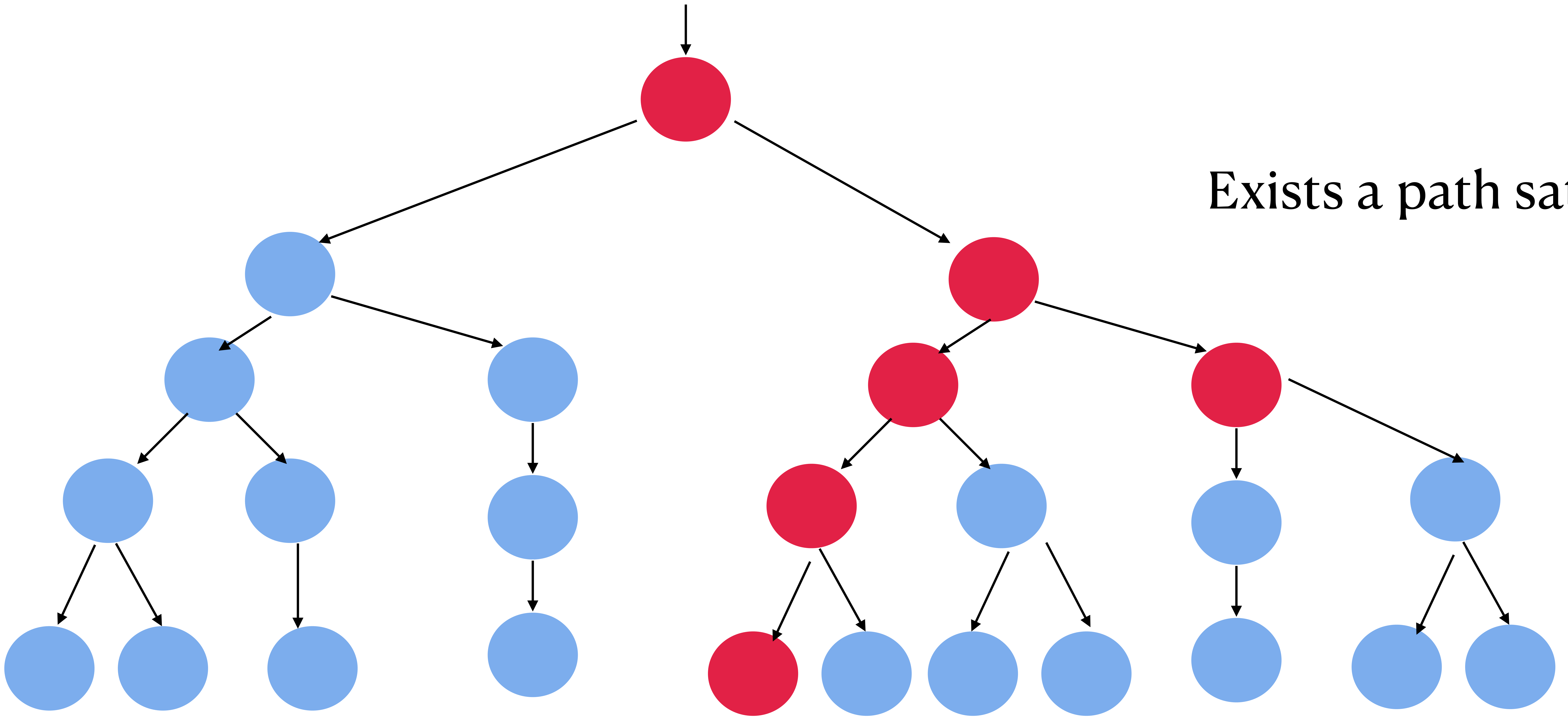
Talks about properties of trees!



Exists a path satisfying $\Diamond(red)$

Computation Tree Logic (CTL)

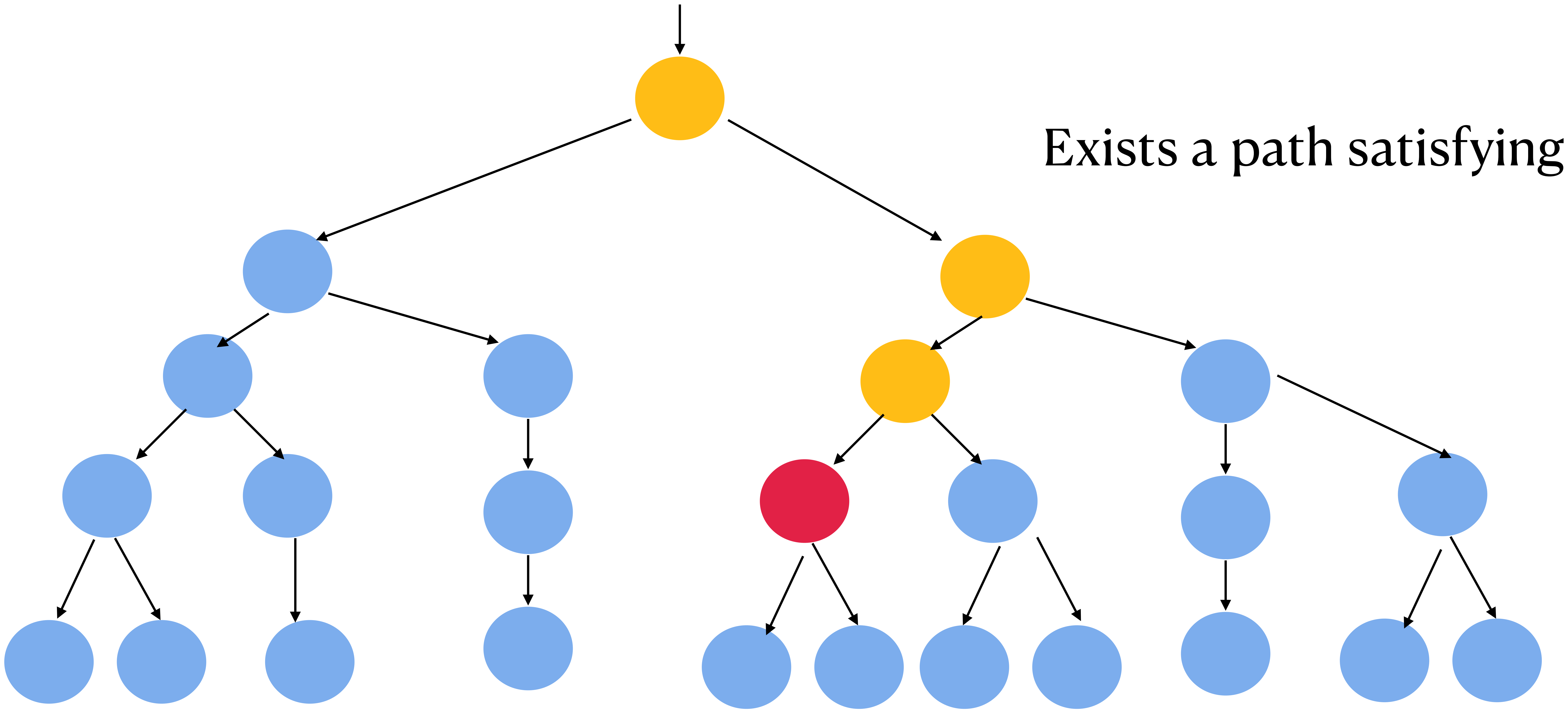
Talks about properties of trees!



Exists a path satisfying $\square(red)$

Computation Tree Logic (CTL)

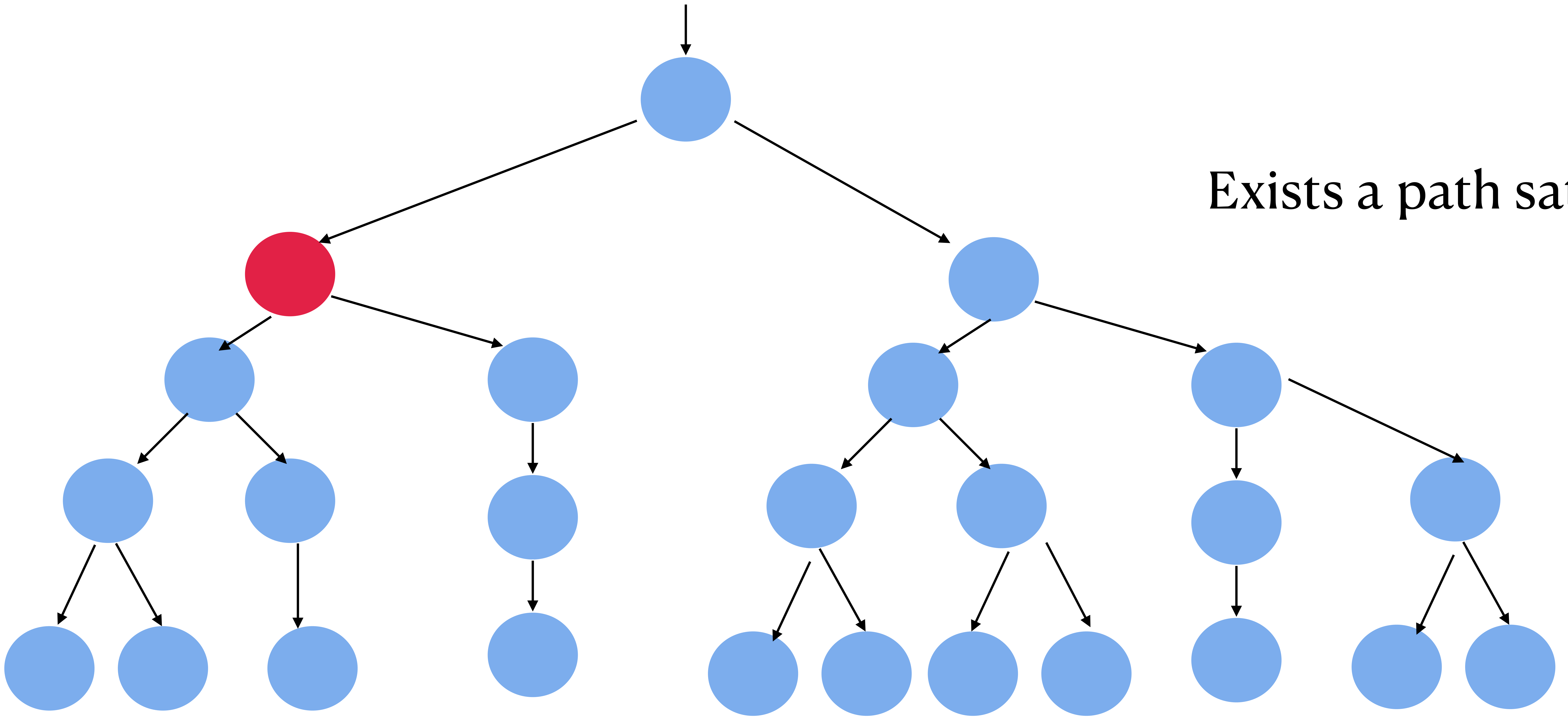
Talks about properties of trees!



Exists a path satisfying (*yellow*) **U** (*red*)

Computation Tree Logic (CTL)

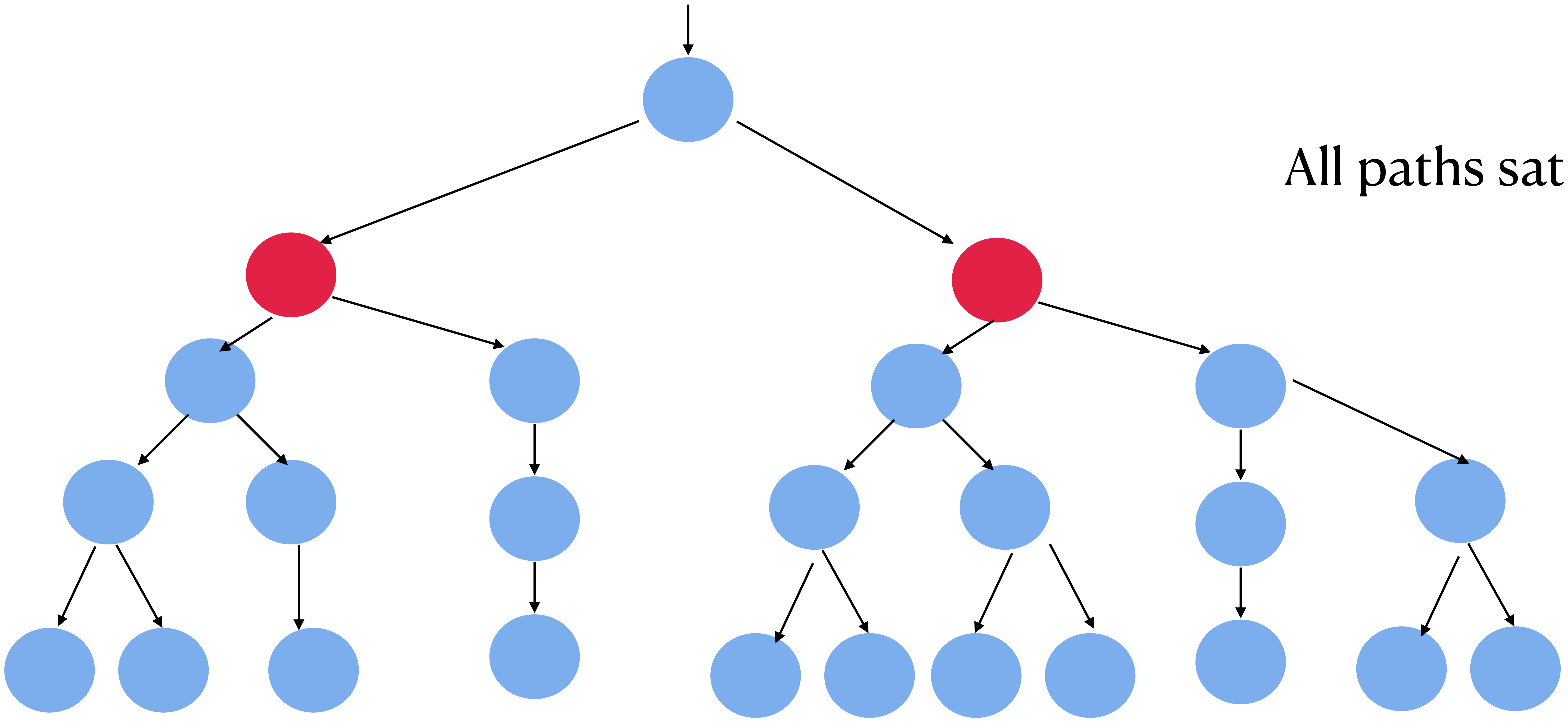
Talks about properties of trees!



Exists a path satisfying $\mathbf{N}(\mathit{red})$

Computation Tree Logic (CTL)

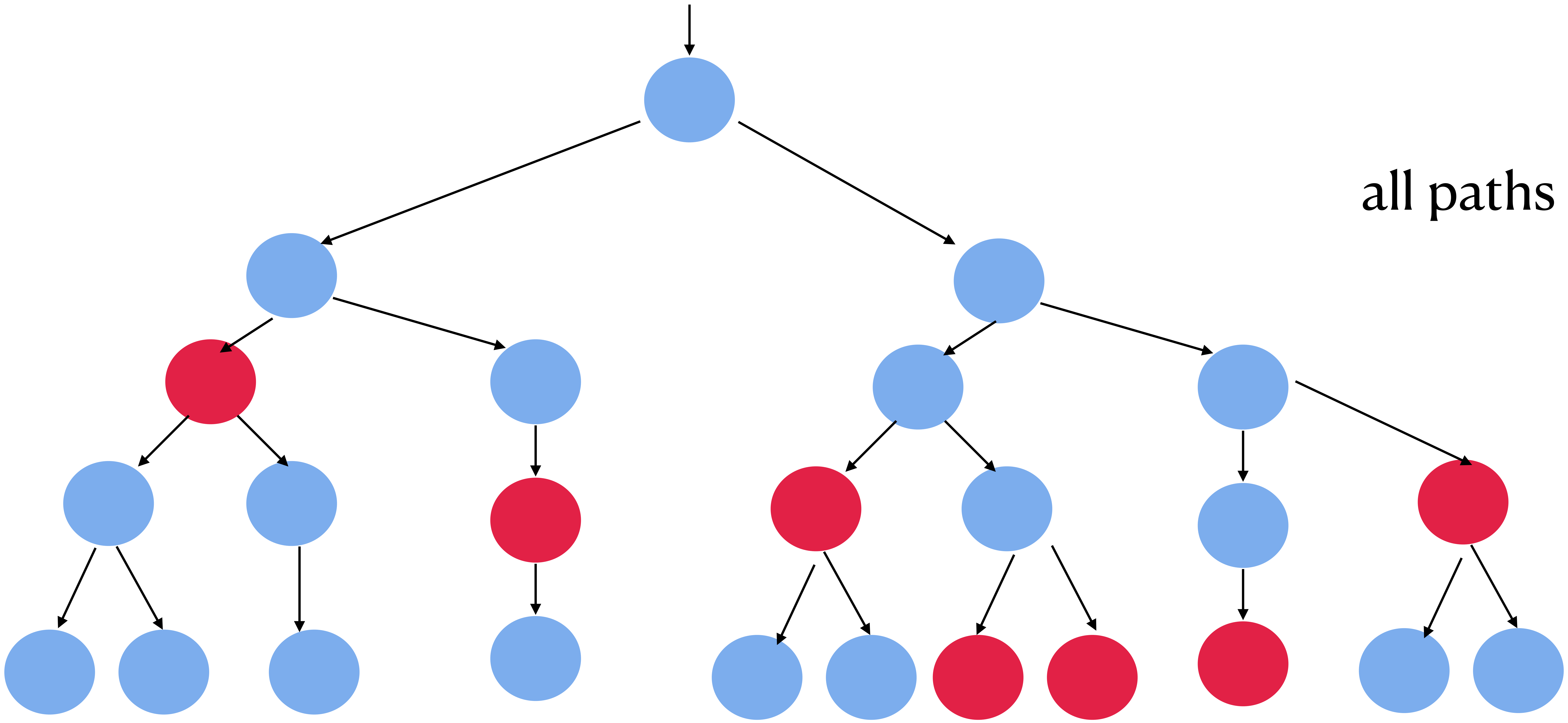
Talks about properties of trees!



All paths satisfy **N**(*red*)

Computation Tree Logic (CTL)

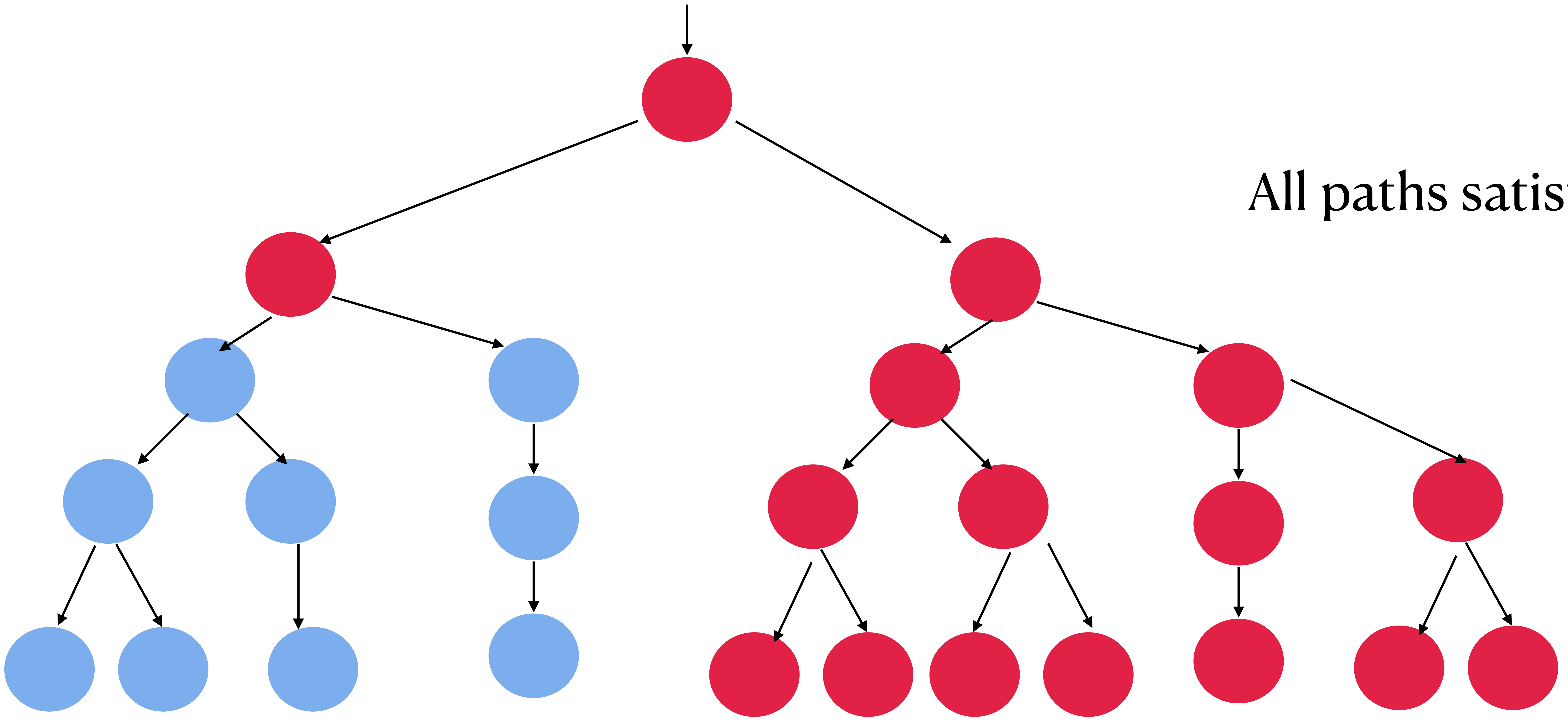
Talks about properties of trees!



all paths satisfy $\Diamond(\mathit{red})$

Computation Tree Logic (CTL)

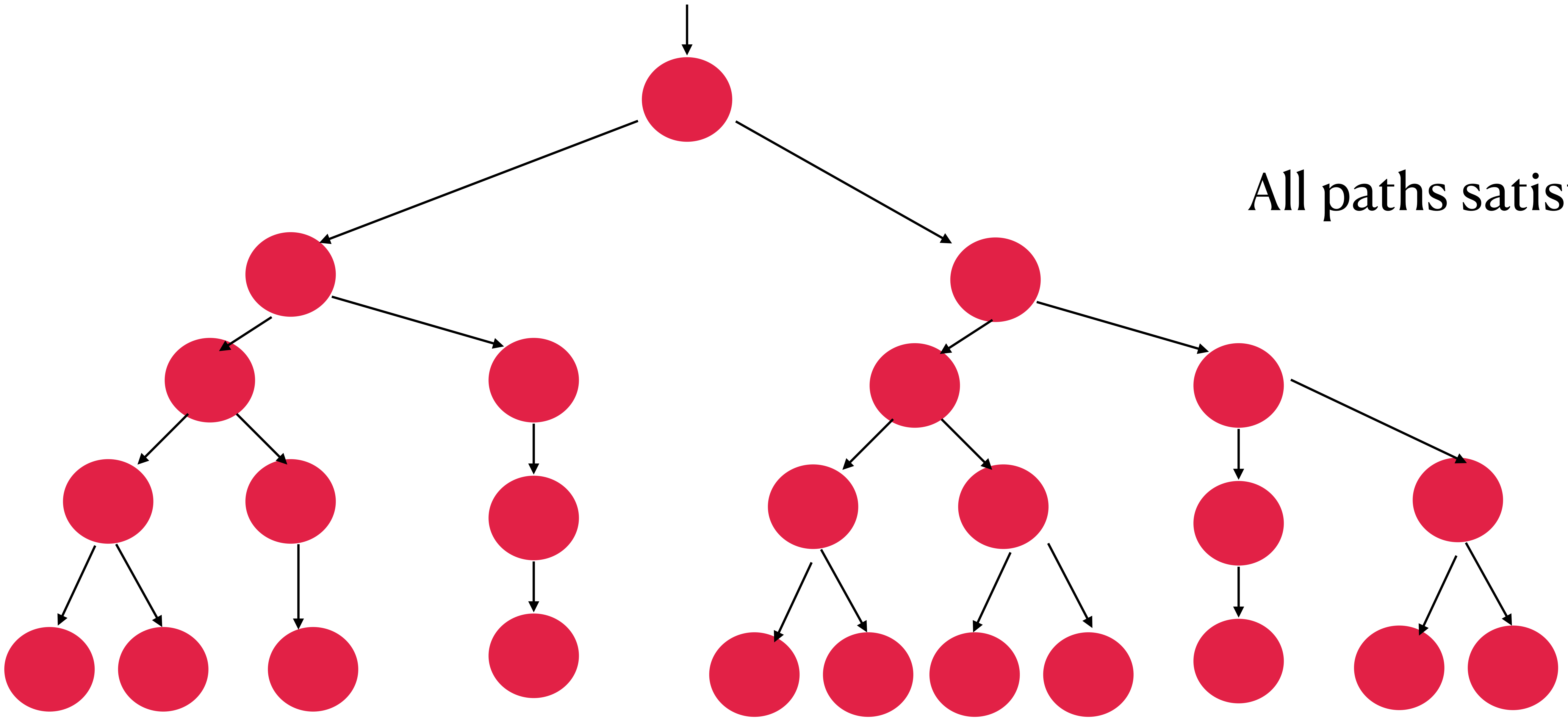
Talks about properties of trees!



All paths satisfy $\square(red)$ ✗

Computation Tree Logic (CTL)

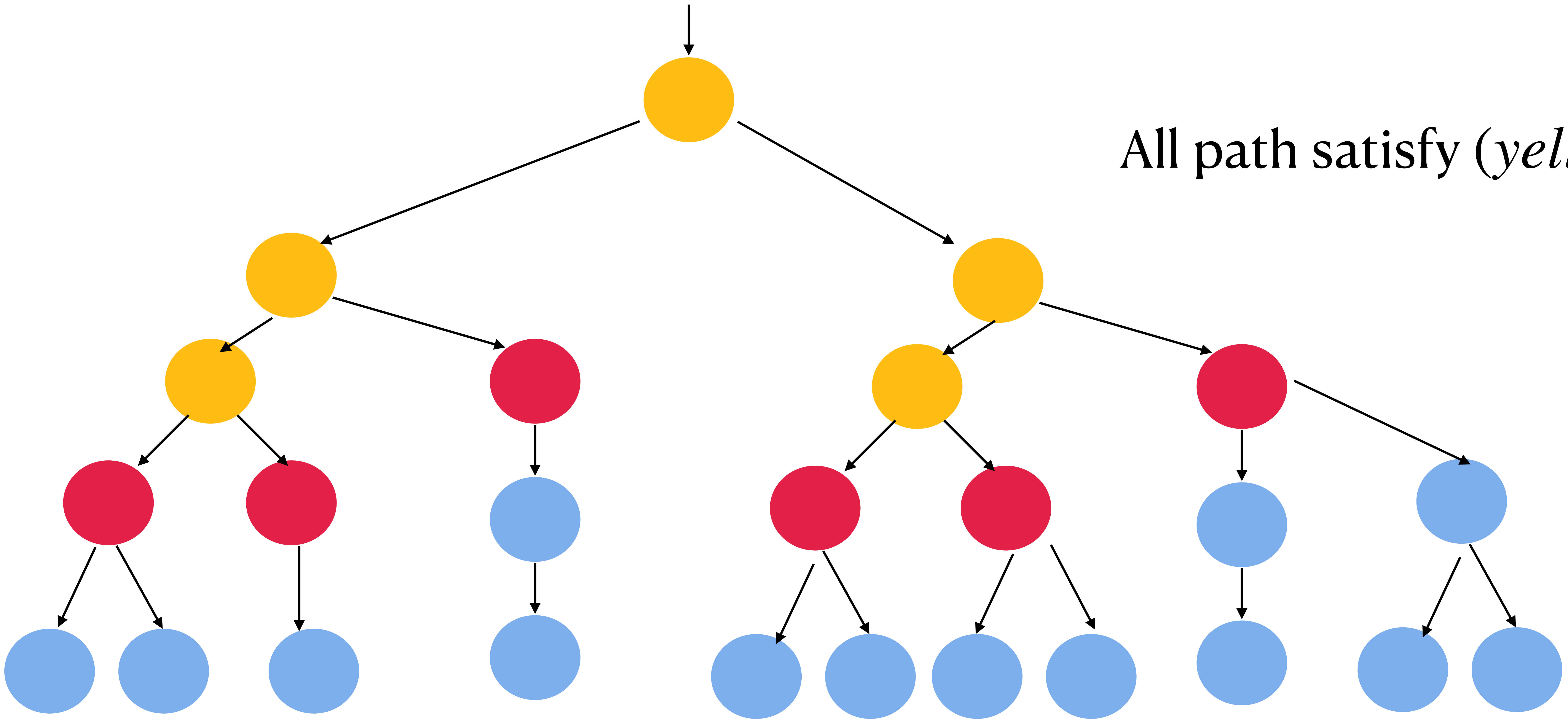
Talks about properties of trees!



All paths satisfy $\Box(red)$

Computation Tree Logic (CTL)

Talks about properties of trees!



All path satisfy (*yellow*) **U** (*red*)

Computation Tree Logic (CTL)

LTL — deals with paths or traces.

CTL — branching time structure (Trees)

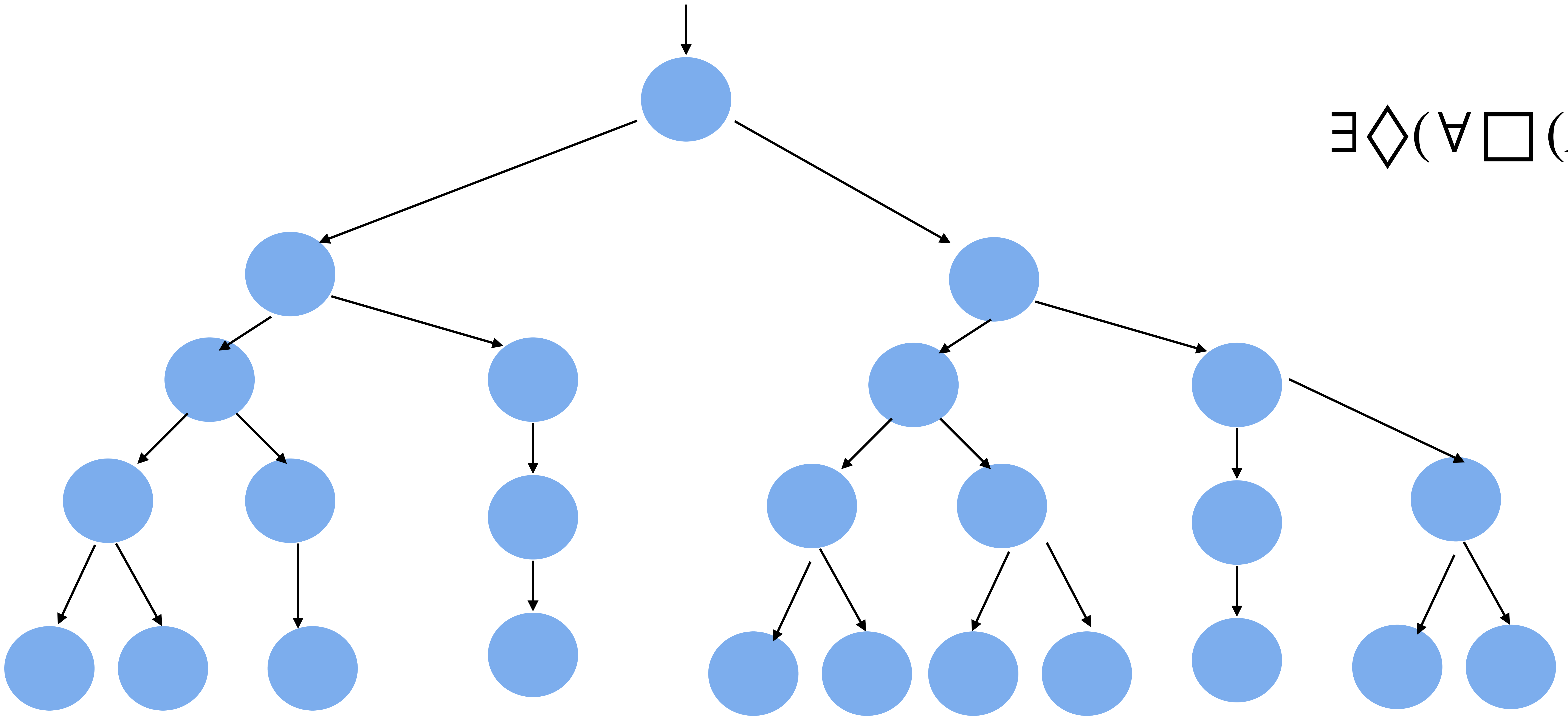
Explicitly introduces path quantifiers!

$\exists^P, \forall^P$ — (in general, we would write as $\exists, \forall$)

$\exists \Diamond red$	$\forall \Diamond red$
$\exists \Box\ red$	$\forall \Box\ red$
$\exists\ yellow\ \mathbf{U}\ red$	$\forall\ yellow\ \mathbf{U}\ red$
$\exists \mathbf{N}\ red$	$\forall \mathbf{N}\ red$

Computation Tree Logic (CTL)

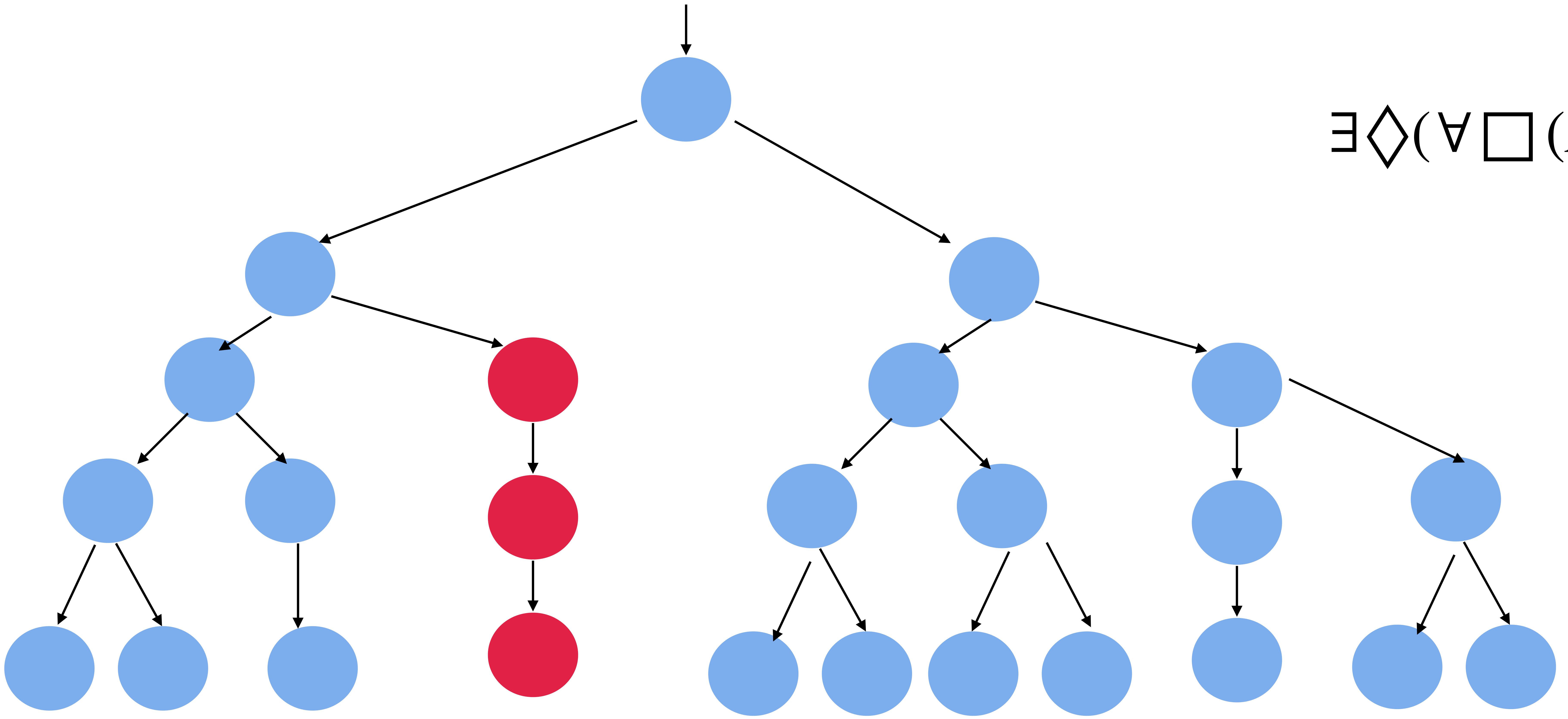
Talks about properties of trees!



$$\exists \Diamond (\forall \Box (Red))$$

Computation Tree Logic (CTL)

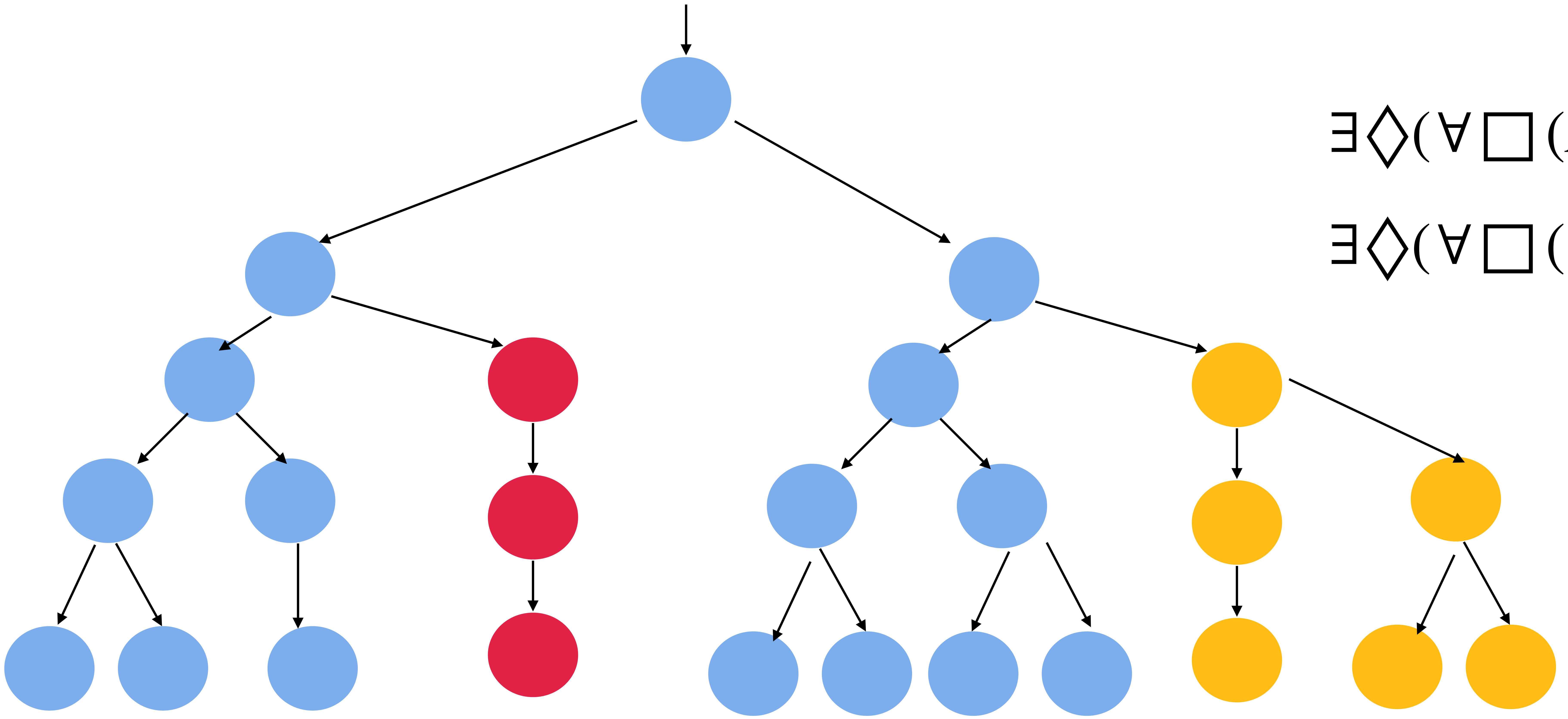
Talks about properties of trees!



$$\exists\Diamond(\forall\Box(Red))$$

Computation Tree Logic (CTL)

Talks about properties of trees!

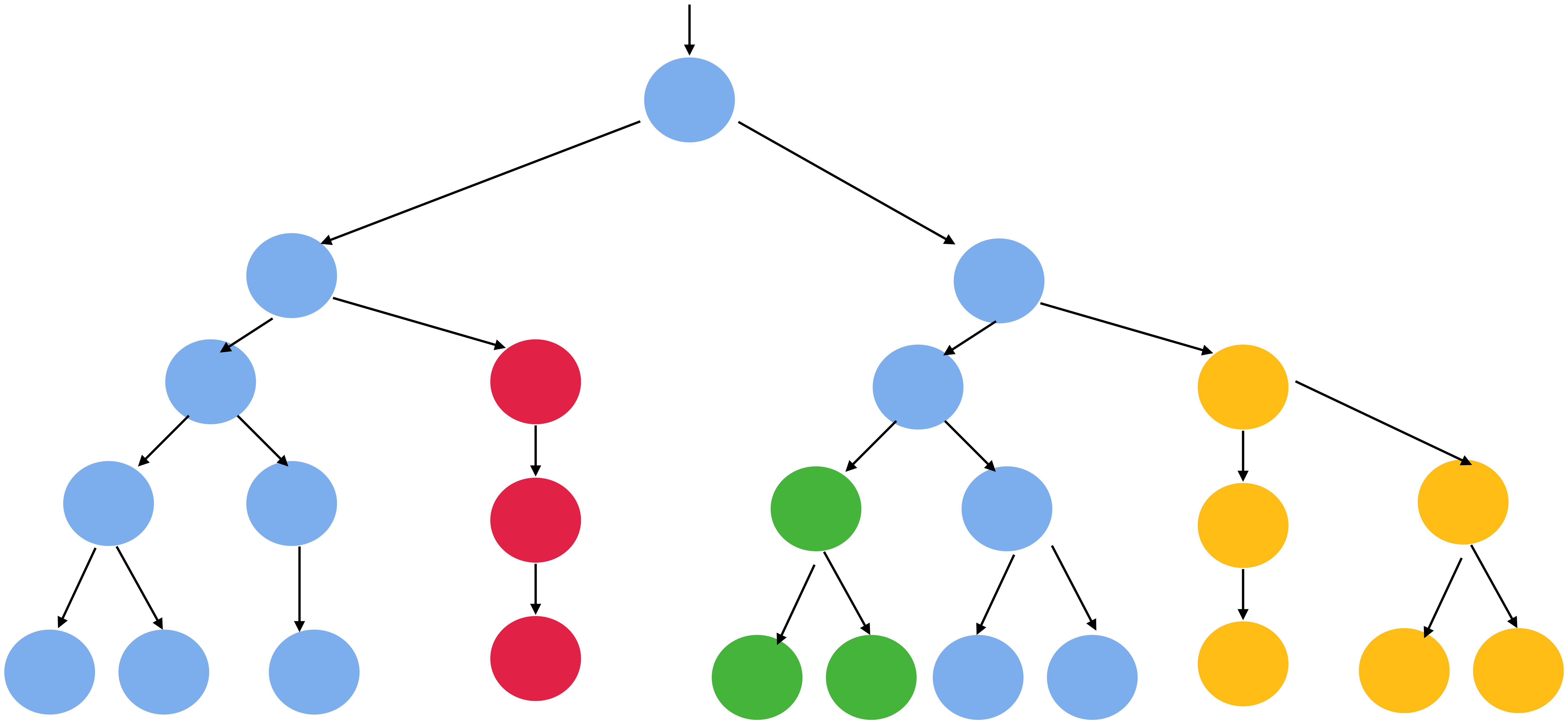


$$\exists\Diamond(\forall\Box(Red))$$

$$\exists\Diamond(\forall\Box(Yellow))$$

Computation Tree Logic (CTL)

Talks about properties of trees!



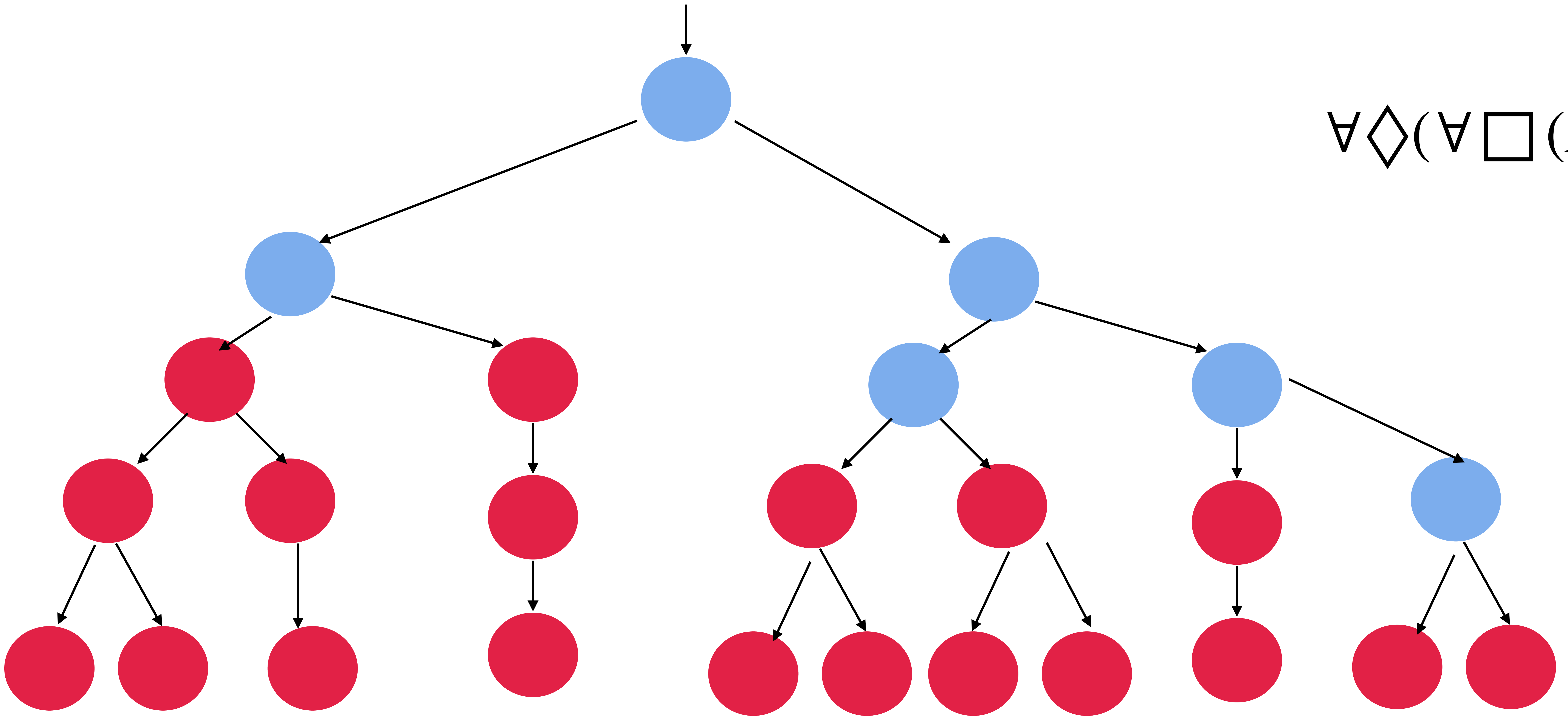
$$\exists\Diamond(\forall\Box(Red))$$

$$\exists\Diamond(\forall\Box(Yellow))$$

$$\exists\Diamond(\forall\Box(Green))$$

Computation Tree Logic (CTL)

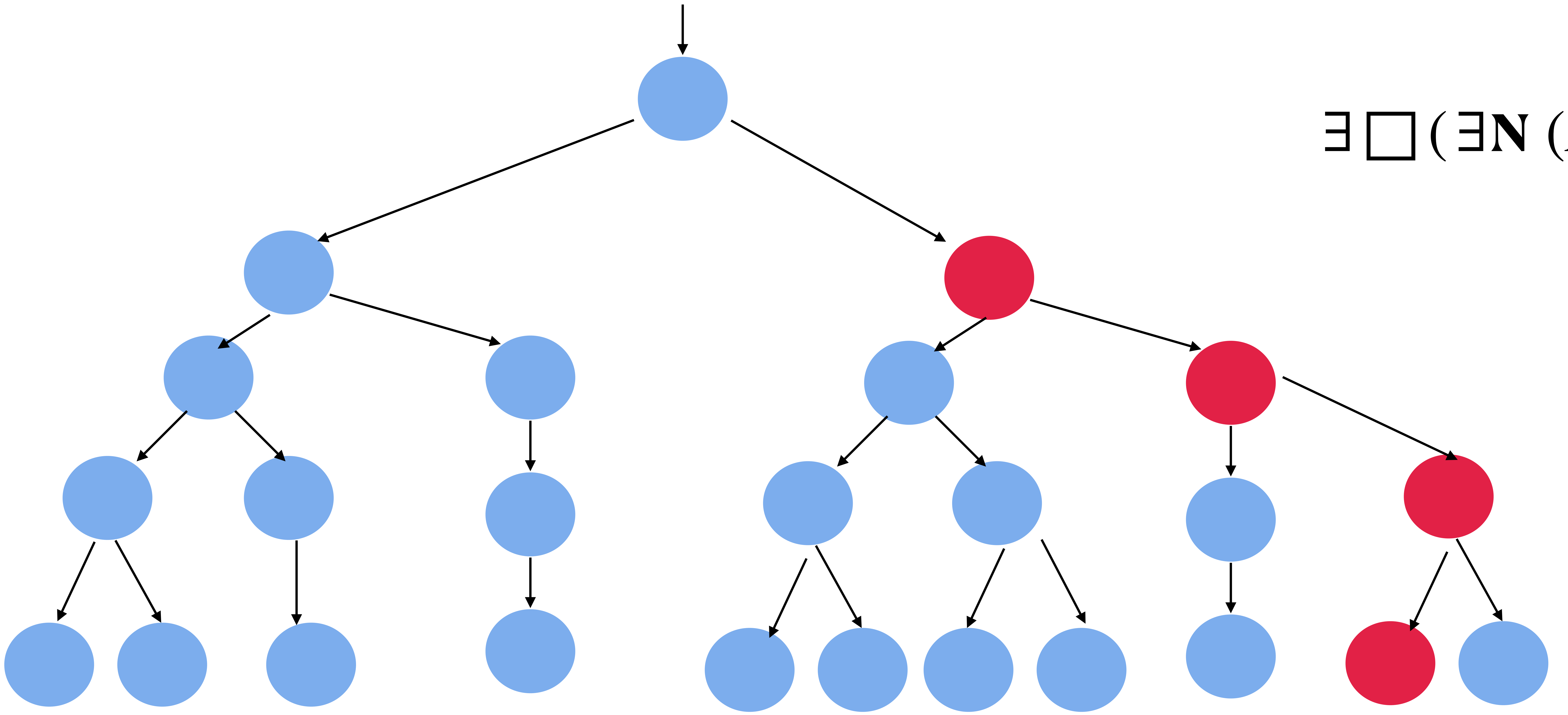
Talks about properties of trees!



$$\forall \Diamond (\forall \Box (Red))$$

Computation Tree Logic (CTL)

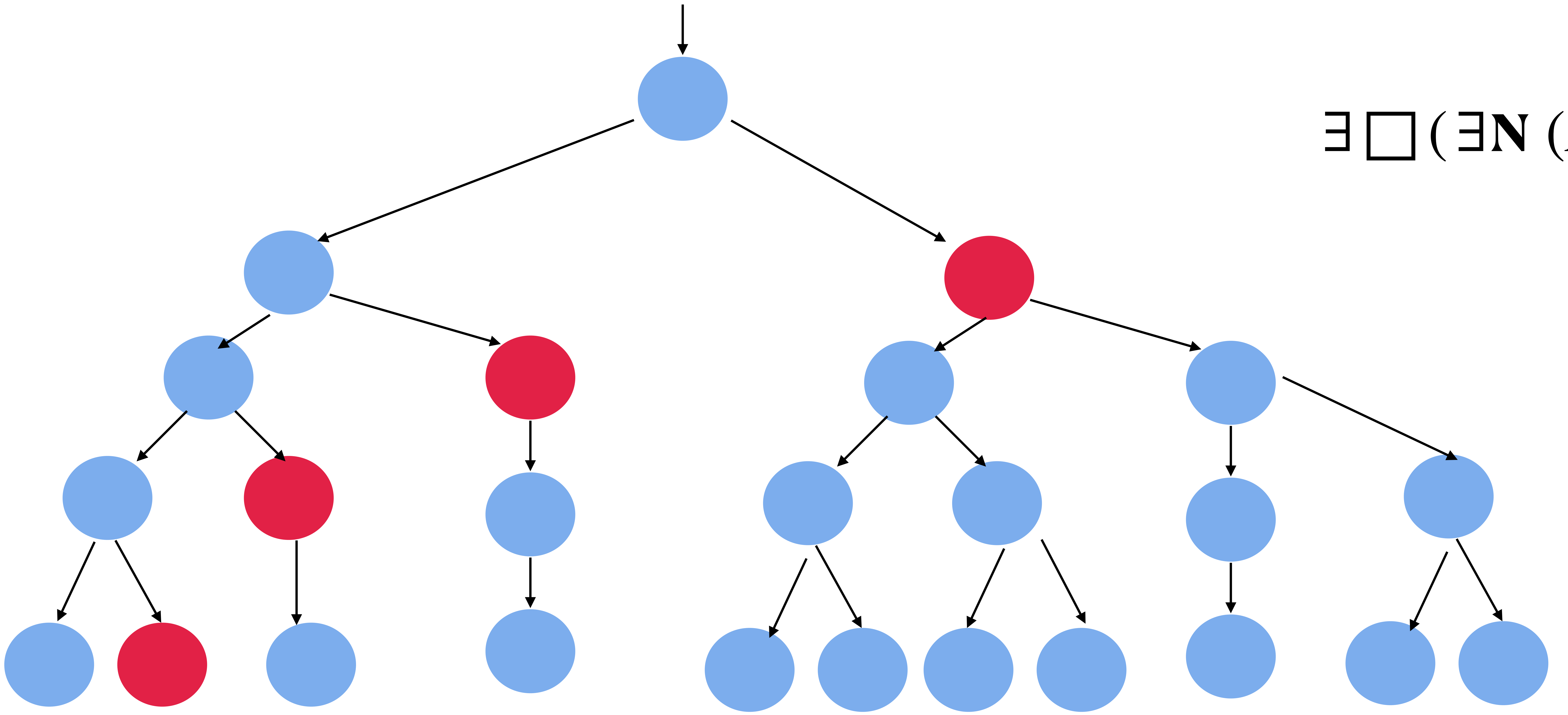
Talks about properties of trees!



$$\exists \Box (\exists \mathbf{N} (Red))$$

Computation Tree Logic (CTL)

Talks about properties of trees!

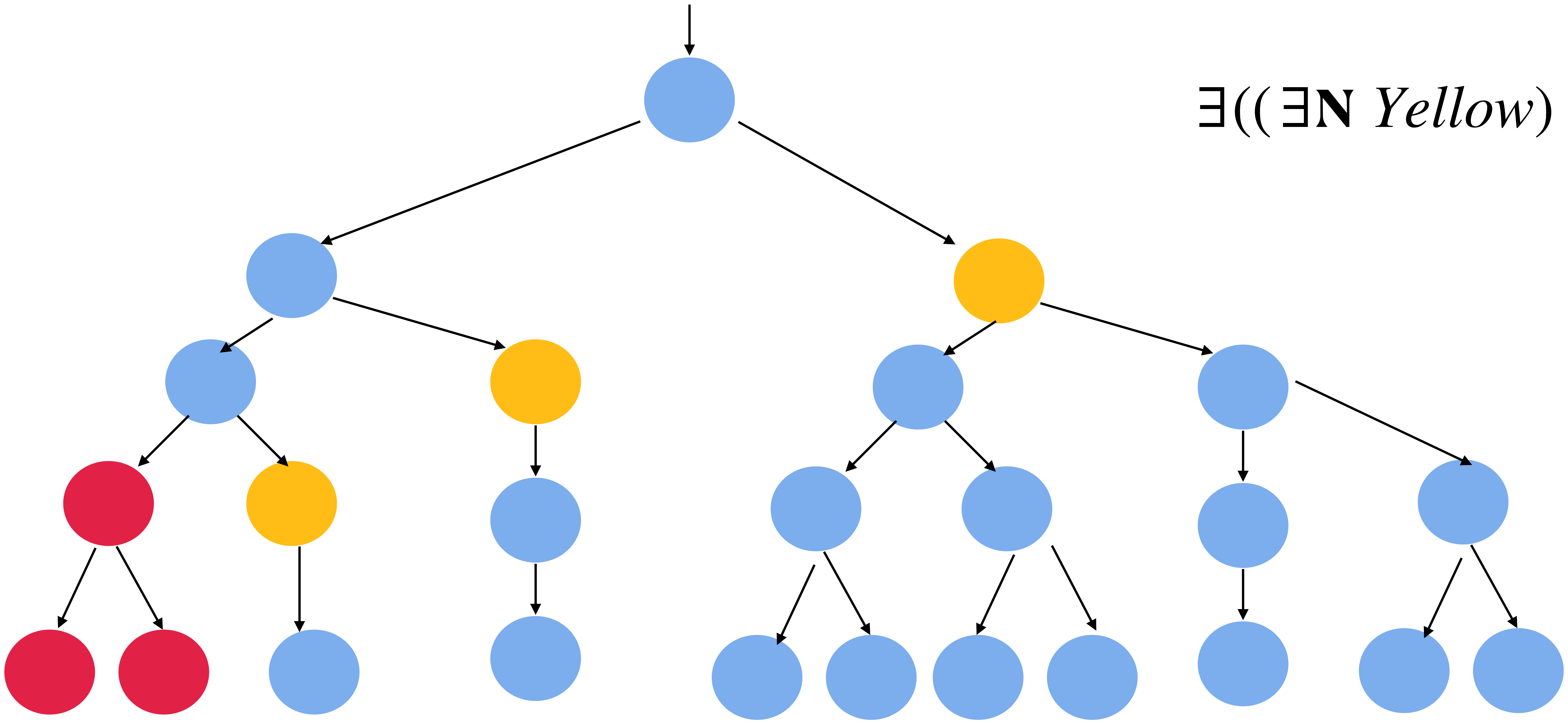


$$\exists \Box (\exists \mathbf{N}\ (Red))$$

Computation Tree Logic (CTL)

Talks about properties of trees!

$$\exists((\exists \mathbf{N}\ Yellow)\ \mathbf{U}(\forall \square (Red)))$$



CTL Syntax

F, F_1 = True |

p (atomic proposition) |

$F_1 \wedge F, F_1 \vee F, F \rightarrow F_1, F_1 \leftrightarrow F$ |

$\neg F$ |

$\forall \mathbf{N}\ F \mid \forall \square F \mid \forall \lozenge F \mid \forall (F\ \mathbf{U}\ F_1)$ |

$\exists \mathbf{N}\ F \mid \exists \square F \mid \exists \lozenge F \mid \exists (F\ \mathbf{U}\ F_2)$

$\exists \lozenge \square F$ Not a WWF!!

$\exists \lozenge (\mathbf{N}\ F)$ Not a WWF!!

CTL : Semantics

Semantics with respect to a given Kripke Structure M

Let $\pi = s_0, s_1, s_2, \ldots$ $\pi(i) = s_i$ State at i^{th} level. $\pi^i = s_i, s_{i+1}, s_{i+2}, \ldots$ Suffix of π

$< M, s_o > \vDash p$ Iff $p \in \pi(0)$ $< M, s_i > \vDash p$ Iff $p \in \pi(i)$

$< M, s_i > \vDash \forall \mathbf{N}\ F_1$ Iff $\forall \pi \in \{s_i, s_{i+1}, s_{i+2}, \ldots, \}$ $< M, s_{i+1} > \vDash F_1$

$< M, s_i > \vDash \exists \mathbf{N}\ F_1$ Iff $\exists \pi \in \{s_i, s_{i+1}, s_{i+2}, \ldots, \}$ $< M, s_{i+1} > \vDash F_1$

$< M, s_i > \vDash \forall \Box F_1$ Iff $\forall \pi \in \{s_i, s_{i+1}, s_{i+2}, \ldots, \}$ $\forall j \geq i, < M, s_j > \vDash F_1$

$< M, s_i > \vDash \exists \Box F_1$ Iff $\exists \pi \in \{s_i, s_{i+1}, s_{i+2}, \ldots, \}$ $\forall j \geq i, < M, s_j > \vDash F_1$

$< M, s_i > \vDash \forall \Diamond F_1$ Iff $\forall \pi \in \{s_i, s_{i+1}, s_{i+2}, \ldots, \}$ $\exists j \geq i, < M, s_j > \vDash F_1$

$< M, s_i > \vDash \exists \Diamond F_1$ Iff $\exists \pi \in \{s_i, s_{i+1}, s_{i+2}, \ldots, \}$ $\exists j \geq i, < M, s_j > \vDash F_1$

CTL : Semantics

Semantics with respect to a given Kripke Structure M

Let $\pi = s_0, s_1, s_2, \ldots$ $\pi(i) = s_i$ State at i^{th} level. $\pi^i = s_i, s_{i+1}, s_{i+2}, \ldots$ Suffix of π

$< M, s_o > \models p$ Iff $p \in \pi(0)$ $< M, s_i > \models p$ Iff $p \in \pi(i)$

$< M, s_i > \models \forall(F \ \mathbf{U} \ F_1)$ Iff $\forall \pi \in \{s_i, s_{i+1}, s_{i+2}, \ldots, \}$

$\exists j \geq i, < M, s_j > \models F_1 \ \& \ \forall \ i \leq k < j, < M, s_k > \models F$

$< M, s_i > \models \exists \ (F \ \mathbf{U} \ F_1)$ Iff $\exists \pi \in \{s_i, s_{i+1}, s_{i+2}, \ldots, \}$

$\exists j \geq i, < M, s_j > \models F_1 \ \& \ \forall \ i \leq k < j, < M, s_k > \models F$

CTL :Examples

Safety: "something bad will never happen"

$$\neg(\exists \Diamond p) \equiv \forall \Box \neg p$$

Reactor_temp is never going to be above 1000.

$$\forall \Box \neg(ReactorTemp > 1000)$$

If car takes left, then immediately car should not take right.

$$\forall \Box \neg(left \ \wedge \ \exists \mathbf{N} \ right)$$

$$\neg \exists \Diamond \neg(left \ \wedge \ \forall \mathbf{N} \ right)$$

CTL :Examples

Liveness: “something good will happen”

$$\forall \Diamond p$$

All students will get their degree

$$\forall \Diamond (Student \wedge degree)$$

If you start something you will eventually finish it.

$$\forall \Box (start \rightarrow \forall \Diamond Finish)$$

CTL :Examples

Correlation: $\Diamond p \rightarrow \Diamond q$ What will be the equivalent CTL formula?

$\forall \Diamond p \rightarrow \forall \Diamond q$ If all the paths have p along them then all the paths have q along them!