

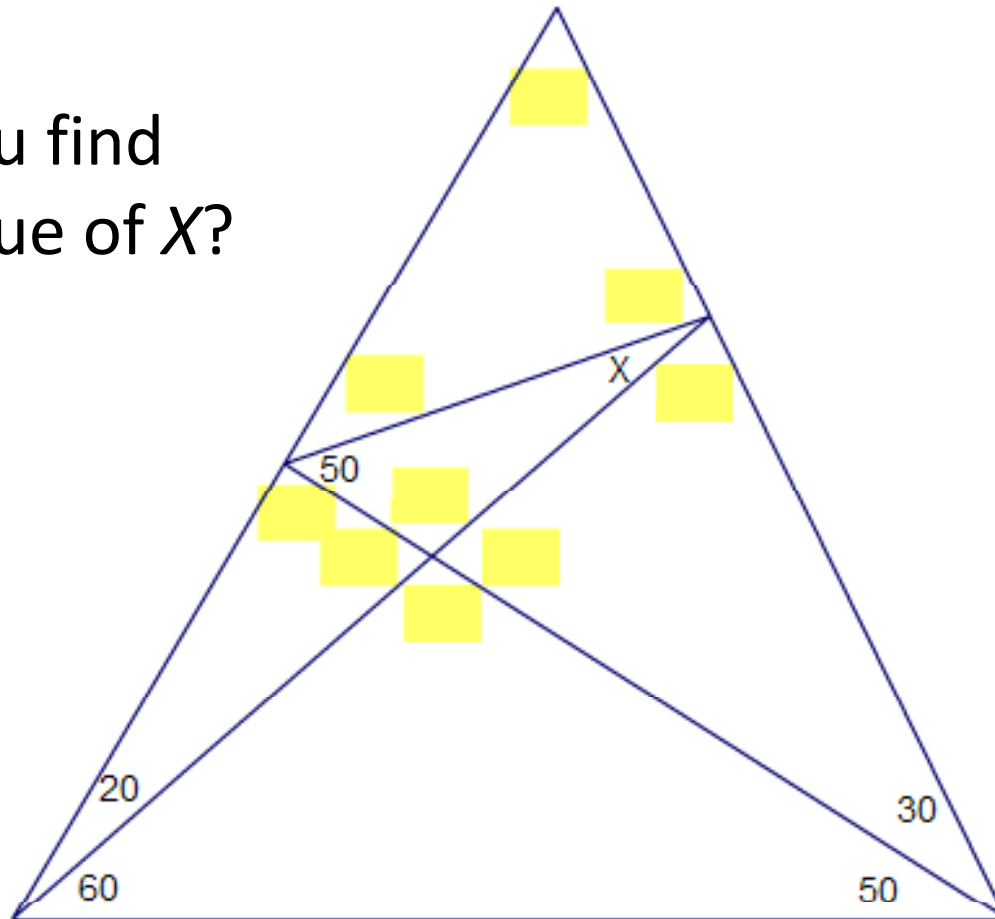
# Building a problem-solving cognitive tutor for geometry

Gustavo Santos, [gsantos@cs.cmu.edu](mailto:gsantos@cs.cmu.edu)

Aaron Wangberg, [awangberg@winona.edu](mailto:awangberg@winona.edu)

# Our Problem

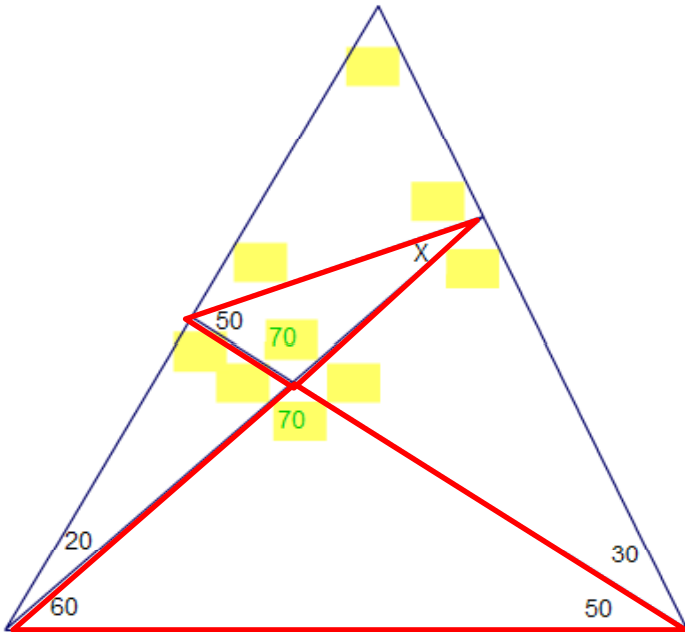
Can you find  
the value of  $X$ ?



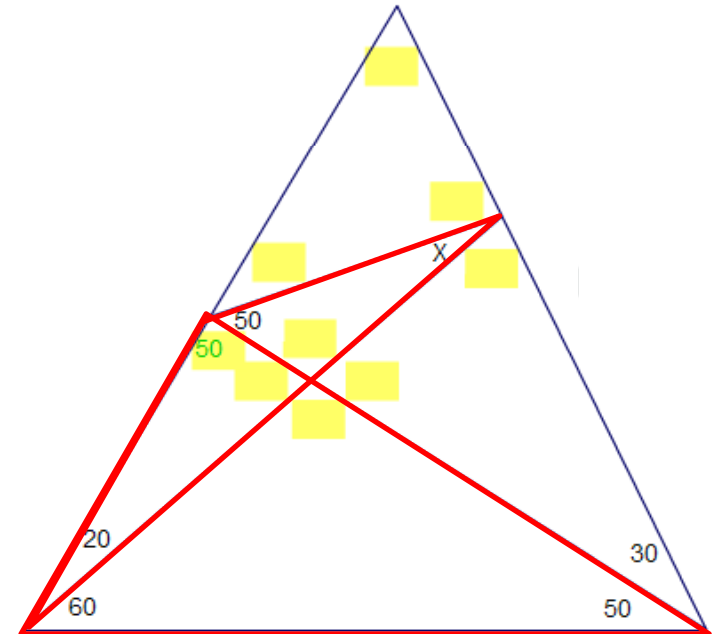
# Our Task

- Develop a problem solving strategy for a problem in which:
  - Steps are semi-ordered
  - There are multiple solution paths
  - Some steps are not necessary
  - Relies upon basic geometry knowledge

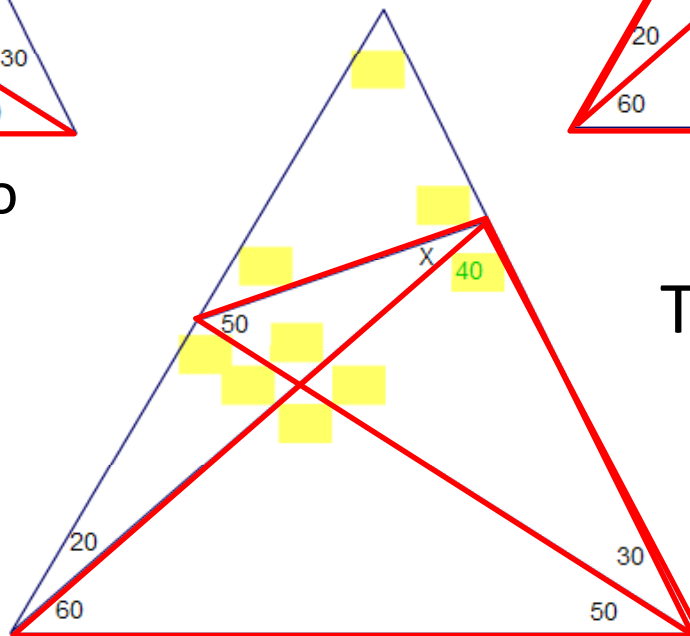
# Three Possible Solutions



One solution uses two  
“small” triangles and  
vertical angles



Two solutions use  
two “large”  
triangles



# Example Tracing Tutor

Connected • AuthorTime • v. 2.3.33

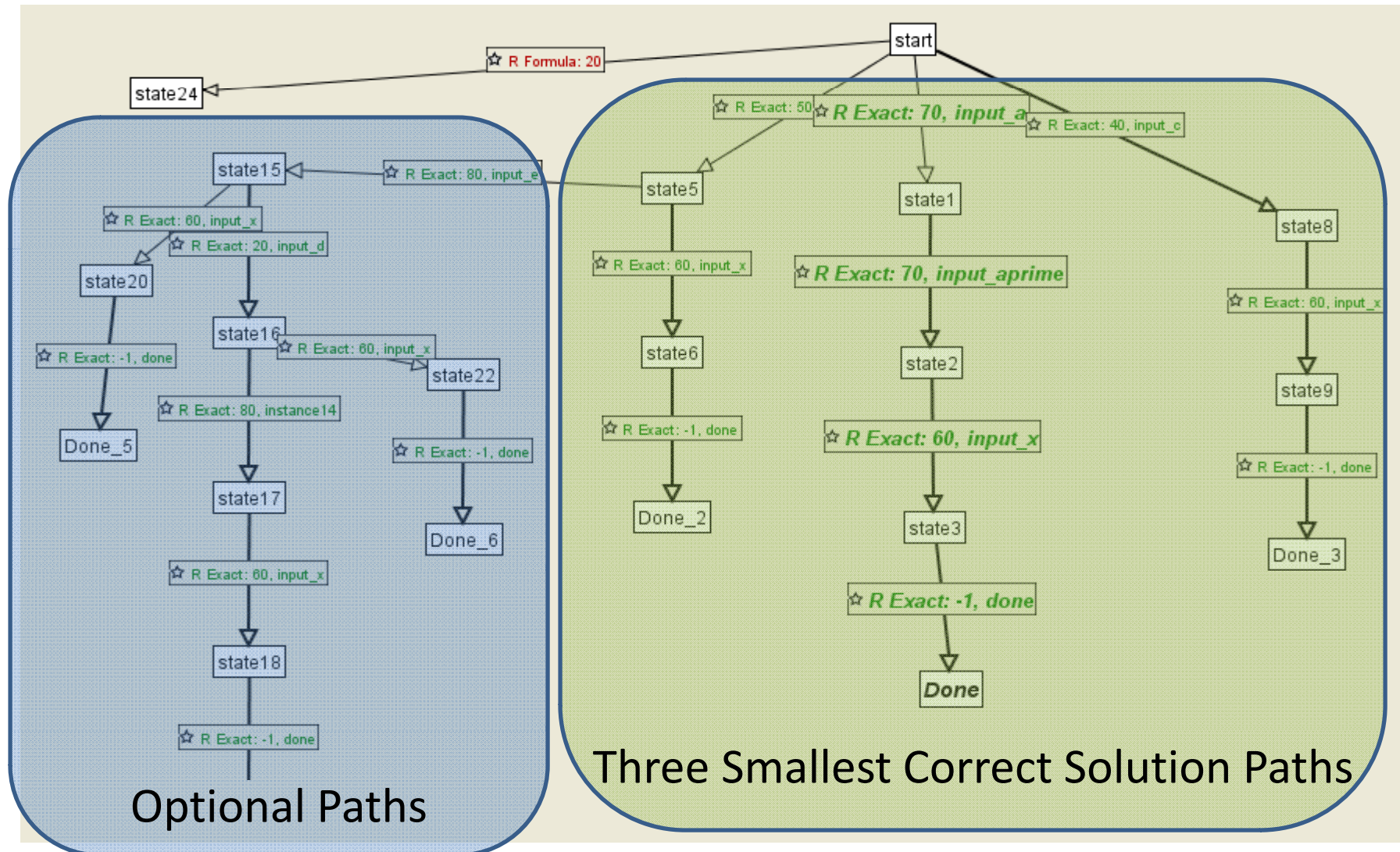
Find the value of the angle X.

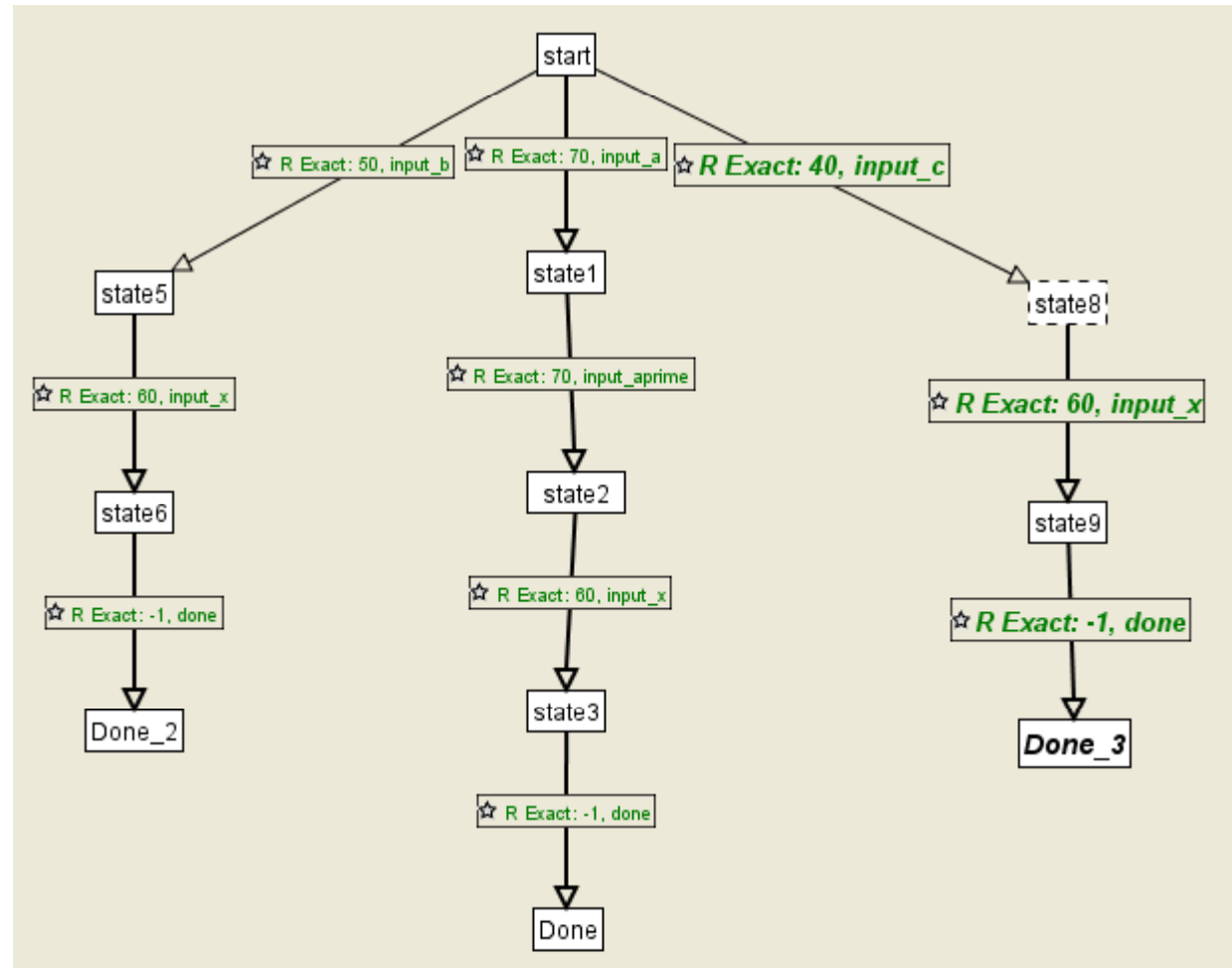
Hint

$x =$

Done

# Behavior Tree





# It gets Complicated!

- The unnecessary steps add complexity
  - Up to 5 steps possible at any node of solution path
  - Steps occur in any order
  - Require up to  $5! = 120$  steps for each node
  - Double this number if providing hints
  - Small solutions tree requires 10 nodes
  - Tutoring tree would require ~2400 nodes

.....Or, allow “optional clusters” of steps!



# Modify the problem -> Simplify Tree

Connected • AuthorTime • v. 2.3.33

1. You want to find the value of the angle  $X$ . How many angles must you find in order to determine  $X$ ?

2. Fill in those angles in the diagram.

3. What is the value of  $X$ ?

$x =$

Done

Hint

The diagram shows a large triangle with interior lines. The bottom-left vertex has an angle of 60. The bottom-right vertex has an angle of 50. The top vertex has an angle of 30. A line from the top vertex to the base creates an angle of 20 at the top-left and 50 at the bottom-right. Another line from the top vertex to the base creates an angle of 50 at the top-left and 30 at the bottom-right. The angle X is the angle formed by these two lines at the top vertex. There are several empty boxes for labeling angles in the diagram.

# One possible simplification

## Modify the Problem...

Ask for, and tutor on, the number of angles that must be found in the solution

Connected • AuthorTime • v. 2.3.33

**Hint**

1. You want to find the value of the angle  $X$ . How many angles must you find in order to determine  $X$ ?

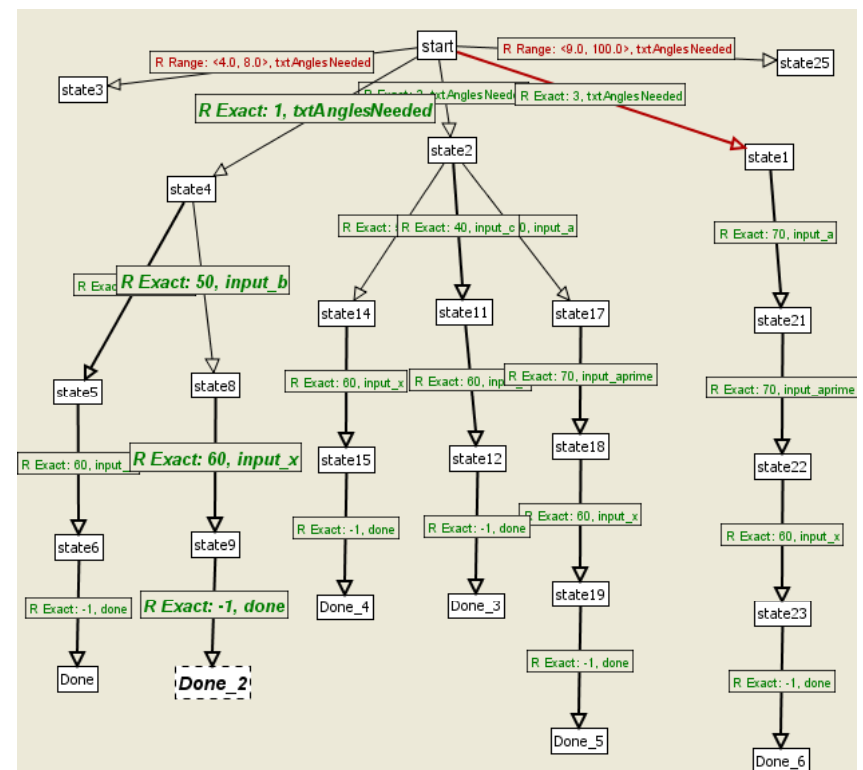
2. Fill in those angles in the diagram.

3. What is the value of  $X$ ?

$x =$

## Create a much simpler tree!

Limit tutoring to student's first answer  
and steps on the small paths



# A Better Solution: Cognitive Tutor

- “We are able to take actions that facilitate learning because we possess a cognitive model of where the student is in the task.” (Anderson, Corbett, Koedinger, Pelletier, *Cognitive Tutors: Lessons Learned*, 1995).
- “Because the system allows students to enter multiple steps leading up to the solution of a problem, it can give immediate feedback” (PSLC Wiki)
- Allows tutoring for similar problems with different initial values

# Working Memory

## Templates:

```
(deftemplate angle
  (slot type_of_angle
    (type String)
  )
  (multislot values_of_the_angle
  )
  (slot value)
)
```

```
(deftemplate triangle
  (slot angle1
    (type ANY)
  )
  (slot angle2
    (type ANY)
  )
  (slot angle3
    (type ANY)
  )
)
```

```
(deftemplate verticalAngle
  (slot angle1
    (type Any)
  )
  (slot angle2
    (type Any)
  )
)

(deftemplate linearAngle
  (multislot values_of_the_angle )
)
```

## Facts:

```
;simple triangles
(bind ?triangle1 (assert (triangle (angle1 ?specAngle1) (angle2 ?specAngle3) (angle3 ?input_a))))
(bind ?triangle2 (assert (triangle (angle1 ?specAngle2) (angle2 ?input_b) (angle3 ?input_f))))
(bind ?triangle3 (assert (triangle (angle1 ?specAngle4) (angle2 ?input_g) (angle3 ?input_c))))
(bind ?triangle4 (assert (triangle (angle1 ?specAngle5) (angle2 ?input_aprime) (angle3 ?input_x))))
(bind ?triangle5 (assert (triangle (angle1 ?input_d) (angle2 ?input_e) (angle3 ?input_h))))

;triangles consisting of 2 triangles
(bind ?triangle6 (assert (triangle (angle1 ?input_b) (angle2 ?specAngle3) (angle3 ?compAngle2))))
(bind ?triangle7 (assert (triangle (angle1 ?compAngle3) (angle2 ?specAngle2) (angle3 ?input_x))))
(bind ?triangle8 (assert (triangle (angle1 ?specAngle4) (angle2 ?compAngle5) (angle3 ?specAngle5))))
(bind ?triangle9 (assert (triangle (angle1 ?specAngle1) (angle2 ?compAngle1) (angle3 ?input_c))))

;triangles consisting of 3 triangles
(bind ?triangle10 (assert (triangle (angle1 ?specAngle2) (angle2 ?compAngle6) (angle3 ?input_d))))
(bind ?triangle11 (assert (triangle (angle1 ?specAngle4) (angle2 ?compAngle4) (angle3 ?input_d))))
```

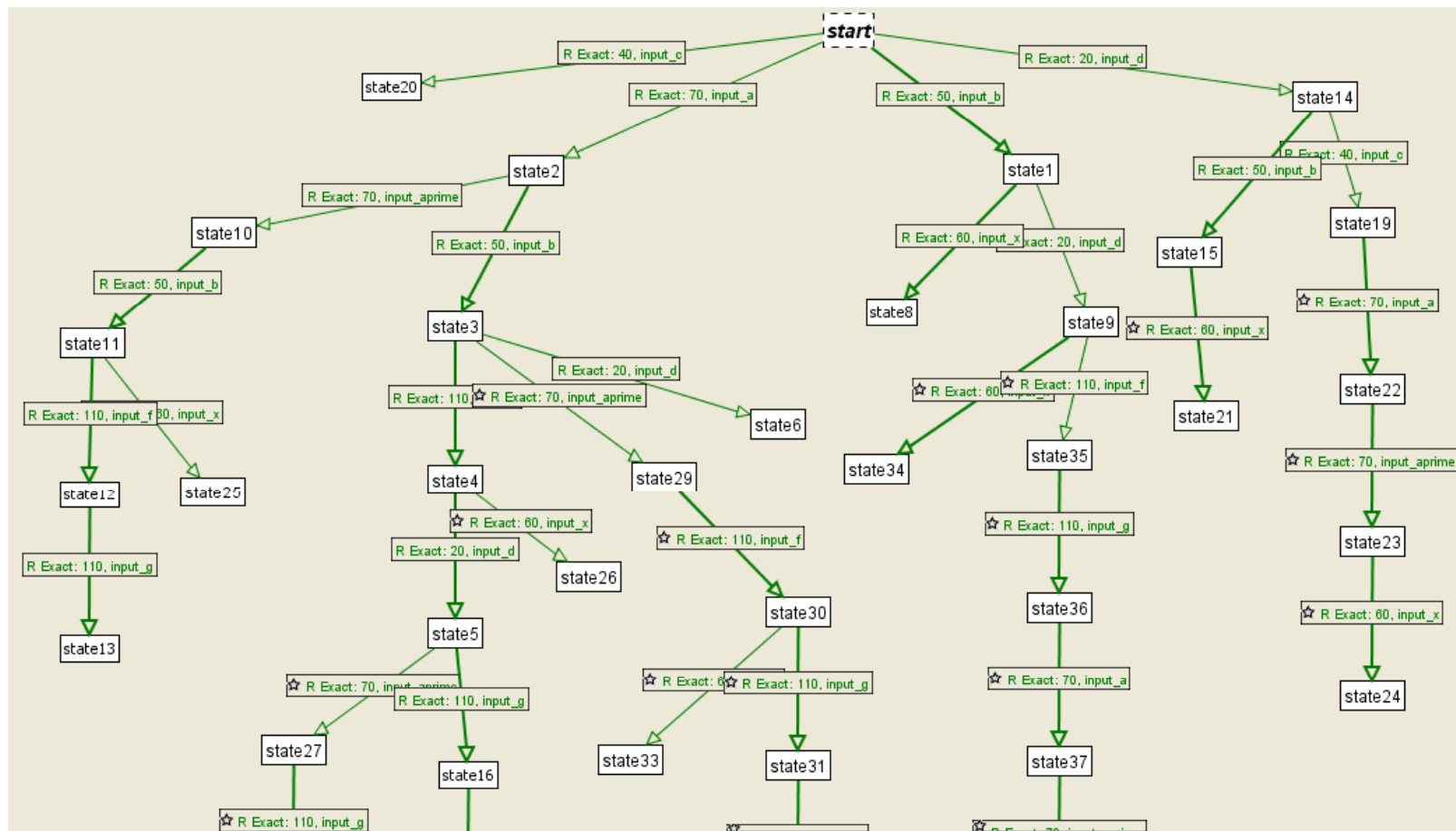
# Rules

```
; If a triangle has two angles with values,  
; set the first angle to 180 - sum of other two angle values.  
(defrule set-first-angle-in-triangle  
  
  ;angle 1 is unknown, angle2 and angle3 are known  
  ?t1 <- (triangle (angle1 ?a1) (angle2 ?a2) (angle3 ?a3))  
  (test (printout t crlf "?t1 = " ?t1 crlf))  
  ;    ?a1 <- (angle (type_of_angle "Pure") (value nil) (values_of_the_angle ?ti))  
    ?a1 <- (angle (value nil) (values_of_the_angle ?ti))  
  ?a2 <- (angle (value ?v2&:(neq ?v2 nil)))&:(> (length$ $?a2-val$) 0))  
  ?a3 <- (angle (value ?v3&:(neq ?v3 nil))) ;&:(> (length$ $?a3-val$) 0))  
  ?ti <- (textInput (name ?selection))  
  =>  
  (bind ?newVal (- 180 ?v2 ?v3))  
  (printout t crlf "?newVal = " ?newVal )  
  (predict-observable-action ?selection UpdateTextField ?newVal )  
  (modify ?ti (value ?newVal))  
  (modify ?a1 (value ?newVal))  
  (set-composite-angle-value)  
)  
  
(defrule set-first-of-pair-vertical-angle  
  ;know one angle, so set other angle  
  ?t1 <- (verticalAngle (angle1 ?a1) (angle2 ?a2))  
  (test (printout t crlf "blah: ?t1 = " ?t1 crlf))  
  ?a1 <- (angle (value nil) (values_of_the_angle ?ti))  
  ?a2 <- (angle (value ?v2&:(neq ?v2 nil)))  
  ?ti <- (textInput (name ?selection))  
  =>  
  (bind ?newVal ?v2)  
  (printout t crlf "blah: ?newVal = " ?newVal)  
  (predict-observable-action ?selection UpdateTextField ?newVal)  
  (modify ?a1 (value ?newVal))  
)
```

## Need rules to determine:

- Third angle in triangle when first two are known
- Equality of vertical angles
- Remaining angle when angle sum is 180 degrees

# (Partial) Cognitive Behavior Tree



Multiple Solution Paths... not just three shortest paths!

# Reflections

- Example tracing tutor can handle problems involving optional steps, but it takes a lot of effort to represent the behavior tree and it is not easy to re-use the represented knowledge tracing
- Cognitive models provide more effective tutoring solutions for problems involving multiple paths or unnecessary steps
- Cognitive models could help tutoring of open-ended problems