Sample Problems and Solutions for Modeling Quiz: State Machine Diagrams
CS 2720

Sample Problems

1. Answer the following questions regarding the diagram (and underlying model) given below.

(a) Give the state the system would be in if the following events happened in the order listed: system starts, invalid sensor reading, invalid sensor reading, driver holds OK, throttle position switch triggered.

(b) Give the actions (activities) the system would perform if the following events happened in the order listed: system starts, driver holds OK, timer expires with a temperature of 150, invalid sensor reading.

(c) Give a sequence of events that would visit every state at least once, including the final state. Where necessary (e.g. a guarded transition), state the relevant system configuration for each event.

(d) Why is setting the 10 minute timer an entry activity of “Open Loop Mode” and not a “do” activity?

(e) If the Fail Safe Mode state had an entry activity of “sound chime”, would the system behavior still be the same?
2. Identify two “diagram smells” in the diagram below and state why they are considered “smells”.

3. Create a state machine diagram for the model summarized below.

Consider the process of how a law is created in the hypothetical country of Andrewica. A law begins as a simple draft. When a law is just a draft, it can be killed by receiving one negative comment. If, however, the draft receives five positive comments, the draft becomes a proposal and a request for comment is sent out to the public.

While waiting for the public to provide comments, a proposal may still be killed if it receives at least twenty negative comments. However, if at least one-third of the public voice support for the proposal, it becomes a bill.

A bill is continuously debated by the council until the council makes one of two decisions: reject or approve. If the council rejects the bill and this is the first time they have done so, it is sent back to the proposal stage and the council suggests changes. If the council rejects a bill for a second time, it is terminated.

If the bill is approved, it is opened for public vote. Once every citizen has voted on the approved bill, it either becomes a law (if and only if the majority supported it) or it is sent back to the proposal stage.
Solutions

1. Answer the following questions regarding the diagram (and underlying model) given below. Below are some sample solutions for the questions given. I have tried to model satisfactory answers for each question – in particular, I have tried to show the level of detail and explanation I would like to see for each question.

   (a) Fail Safe Mode
   (b) Set 10 minute timer (entry of Open Loop Mode), sound chime (transition from Open Loop Mode to Fail Safe Mode), repeatedly scroll error message (Fail Safe Mode)
   (c) System start, timer expires and the temperature is 190, invalid sensor reading, invalid sensor reading, error reset signal, driver holds OK, timer expires with a temperature of 110
   (d) We only want to set the timer when we begin open loop mode. If setting the timer was a “do” activity, it would be set repeatedly and never expire, meaning you could never get to Closed Loop Mode.
   (e) No. Currently no chime sounds when going from Warning Mode to Fail Safe Mode. If the chime was an entry activity to Fail Safe Mode, we would sound a chime when going from Warning Mode to Fail Safe Mode.

2. Identify two “diagram smells” in the diagram below and state why they are considered “smells”. As with the other diagrams, there are numerous “smells” here. Note there may be other smell “patterns” possible which are not covered here, but might be on a quiz.

   • Researching has no outgoing transitions and so it is a dead-end state. This means I can never terminate once I begin researching, which is impossible.
   • Relaxing has no incoming transitions and so it is an unreachable state. I have created a state in my model but I have no way to ever reach this state, meaning it is either not needed or I am missing a transition.
   • When in Asleep, if the alarm sounds and the kids are awake on a weekday, there are two different states I could go to. This means my state machine is nondeterministic, which makes it difficult if not impossible to properly review behavior.
   • When in Driving, if I reach work there are two states I could go to (teaching or researching). This means the machine is nondeterministic, as the same sequence of events could end in two different states, making it hard to analyze the system behavior.
   • The transition from Getting Ready to Researching has no event. Either we missed the event that causes this transition, or I have nondeterministic behavior (the same sequence of events could result in more than one different ending states), which is difficult to review.
   • The “do” activity of Getting Ready is not something that will probably be done repeatedly while getting ready, it only needs to be done once (or zero times in the case of balding middle-aged software engineering instructors).
   • There is no final state, meaning as listed my system will never stop. This is likely not what was intended.
3. Create a state machine diagram for the model summarized below.