Remembering

1. Define the following terms:
   - Model
   - Diagram
   - REST architecture

2. What is the difference between a domain model and a design model?

3. Give one reason Sommerville gives for why model-driven architecture has not been a “mainstream approach” to software engineering.
Understanding

4. Answer the following questions regarding the model described by the state machine diagram 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?
5. Answer the following questions regarding the model described by the sequence diagram below.

(a) Describe what the three “steps” (interactions started with outgoing messages) of the Website do in our model – that is, summarize at a higher level than just stating what the UML notations are.

(b) How would the behavior of our model differ if printKey were a synchronous message (instead of the asynchronous message it currently is)?
Analyzing

6. Give two activities where the use of a model may be helpful. Include a description of how the model would be helpful over some other alternative.

Evaluating

7. Why is it important for a software engineer to be familiar with architectural and design patterns?