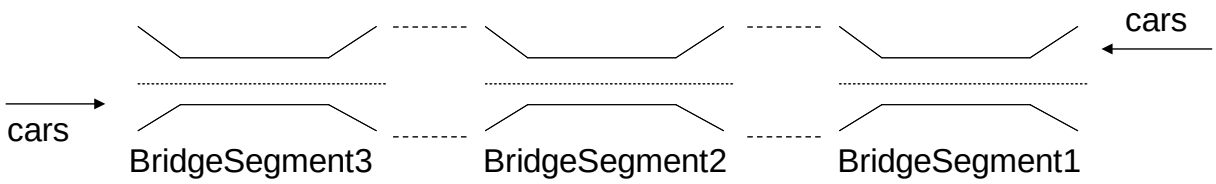
CSC 8840 Homework 1
Spring 2013
Dr. Xiaolin Hu

DEVSJAVA Chapter 4, Exercise 2 (page 64)

- Write an atomic model for the "Bridge Segment"
- Besides the "Bridge Segment" atomic model, also write two "car generator" atomic models (one to generate cars coming from the west, and the other to generate cars coming from the east), and then couple the three models together to test your "Bridge Segment" model.
- Finally, couple the car generators and the three "Bridge Segments" as shown below to form a traffic system. (assume the rode segments between two bridge segments have zero travel time and infinite capacity.)
- You need to package all your source code into a package hierarchy named "Homework2013.BridgeSegment.XXX", where XXX is your first name initial plus last name. For example, if your name is Haidong Xue, the package name should be Homework2013.BridgeSegment.HXue. Note that to make the package work, you will need to create a directory Homework 2013, and then a directory BridgeSegment under Homework2013, and then HXue under BridgeSegment.
- Besides the source code, each student also needs to turn in a document (no more than two pages) to describe the design of your "Bridge Segment" model. This should cover the basic logic of your deltext(), deltint(), deltcon(), and out() functions. Your description can be a informal English description of the model design, or a state diagram (similar to those in the DEVS class slides) with explanations. We will use this document to help grading your model. Note that a formal and detailed description of the model behavior is not needed because we already have your source code. Place the model description document inside your source code directory.
- Turn in your DEVSJAVA code and your model description document by zipping your source code directory (e.g., the direction of HXue in the above example) and emailing the zip file to 2013Spring8840@gmail.com.
- Due date: Feb. 25, 2013



More information:

Two Approaches

1. Model the traffic light and bridge segment together.
2. Model the traffic light separately. In this case, each bridge segment is a coupled model with a bridgeSegment and a traffic light.

If you like, you may program your logic in a stepwise fashion (your final code should finish step2):

1. Step 1: As long as the traffic light is green, a vehicle can move ahead. If the traffic light changes to red when a car is still on the bridge, the cars on the other direction should wait until this car finishes.
2. Step 2: A vehicle can go through the green traffic light only when at least 10 seconds is left before the traffic light changes to red.