



Producer/Consumer Simulation Program

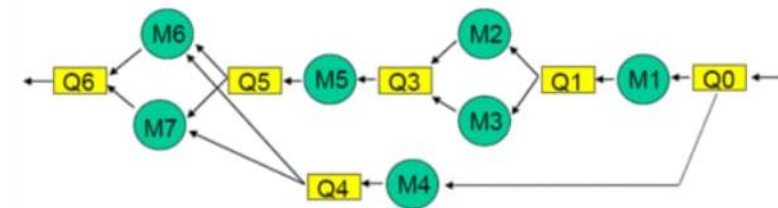
Objectives

Upon completion of this assignment, you will be able to:

- Design an object-oriented queueing simulation program.
- Draw a UML class diagram that represents your model.
- Apply different design patterns as concurrency design pattern, snapshot DP, and observer DP.

Description:

An assembly line that produces different products consists of different processing machines Ms that are responsible for processing the product at different stages and queue Qs to handle product movement between different processing stages (see figure below). In this assignment we will develop a simulation program to simulate this production line as a queueing network.



Requirements:

The simulation system should support the following features:

- Users can graphically add Qs and Ms, connect them via UI arbitrarily.
- The input (products arriving at Q0 to get processed) has a random input rate.
- Each M has a random service time and can serve one product at a time. Once completed, it needs to check the queue if waiting products need to be consumed, if not then it needs to register itself to the queue as ready (Hint: check observer design pattern)
- Each machine is running and processing its products on a separate thread different from other machines' processing threads
- The UI shows the simulation by displaying the number of elements in the Qs in real time.

- Ms flash when they finish servicing an item and every product has its own color (a random color) that will keep it from start till the end and each machine will change its color the product's color being processed by it then change back to a default color once done to make following the simulation easy for the user.
- After the simulation ends, the user can start a new simulation or replay the previous simulation (hint: check snapshot design pattern)
- For simplicity of random generation, it is not required to get into Poisson and exponential distributions.
- For simplicity queues do not overflow.

Deliverables

- You should work individually.
- Develop this assignment in Java spring boot and Vue.js.
- You should provide implementation for the given requirements.
- You should apply the following design patterns appropriately: concurrency DP, snapshot DP, and observer DP.
- You should deliver a report that:
 - Describes thoroughly a full list of the steps required to run your code.
 - Includes a UML Class diagram that represents your model, showing all the classes, attributes and methods.
 - Describes thoroughly how you have applied the required design pattern in your code.
 - Includes snapshots and sample runs of your application running and a simple user guide that explains how to use your application.
 - Includes any design decisions that you have made should be listed clearly.
- Upload your **report**, and **source code zipped** to Microsoft teams.
- Delivering a copy will be severely penalized for both parties, so delivering nothing is so much better than delivering a copy.

Good luck isA 😊