ETEC3702 – Concurrency
Lab 4 – Condition Variables for Selective Waiting

Due date: 9 February by the end of the day.

You are to write a system for simulating an automobile production facility. The production process consists of three sequential stages of production: Fabrication, Painting, and Finishing. There are two Fabrication assembly lines that must both feed one painting station. The cars that come out of the painting station are subsequently sent to the finishing station. There is only enough room in the painting station for 4 vehicles at a time and only enough room in the finishing station for 4 vehicles at a time. All stations can operate concurrently.

We want to track the sequence of events for the entire process of manufacturing runs of 10, 20, 30, and 40 vehicles. Thus we have a set of producer/consumer relationships as follows:

```
Fab1 ----\
          > ---- >  Painting ----->  Finishing ----> Completed
Fab2 ----/
```

Fab1 and Fab2 produce cars to be consumed by the painting station - if all the painting cells are full then the Fab assembly lines should stop and wait. The Painting station then sends it's output to the Finishing station. If the Finishing station's 4 cells are full then the Painting process should stop. Once the Finishing is done, the cards will move to completed.

Your program should print out messages as it processes the vehicles so that production can be tracked. Be sure to include messages telling when a station is full and when a station has halted due to a wait condition.

Example output might look like this:

```
Fab1 produced Car1
Fab2 produced Car2
Fab2 sent Car2 to painting queue
Fab1 sent Car1 to painting queue
Painting queue received Car2
Painting queue received Car1
Painting sent Car2 to Finishing
Finishing received Car2
Finishing sent Car2 to Completed
Painting sent Car1 to Finishing
Finishing received Car1
Finishing sent Car1 to Completed
Fab1 produced Car3
Fab1 sent Car3 to painting queue
Fab1 produced Car4
Fab1 sent Car4 to painting queue
Fab2 produced Car5
Fab2 sent Car5 to painting queue
Painting queue received Car3
Painting queue received Car4
Painting queue received Car5
Fab1 produced Car6
Fab1 sent Car6 to painting queue
Fab2 produced Car7
Fab2 stopped - painting queue full.
Painting sent Car3 to Finishing
Fab2 started
…
```

Use the monitor concept to control access to the Painting and Finishing queues (note that the two fab processes will place cars into the painting queue and cars will be consumed from the painting queue and placed into the finishing queue. Cars will be taken from the finishing queue and then recorded as completed. Use a condition object along with wait and notify statements to implement the selective waiting needed.