Carleton University
Department of Systems and Computer Engineering
SYSC 3303A - RealTime Concurrent Systems - Winter 2026

Assignment 1 — Java Threads

The Autonomous Drone Assembly Line. (This problem was first published as the cigarette smokers problem by S. Patil in 1971, and is one of several classic process coordination problems that are used to evaluate facilities for synchronizing concurrent threads and processes.)

Consider a system with three drone technician threads and one resource agent (assembler) thread.

Each technician wants to assemble an autonomous drone and mark it as ready for deployment. However, to assemble a drone, the technician needs three components:

- Frame,
- Propulsion Unit,
- Control Firmware.

One of the technician threads has an infinite supply of frames, another has an infinite supply of propulsion units, and the third has an infinite supply of control firmware.

The agent (assembler) thread has an infinite supply of all three components. The agent randomly selects two of the three components and places them on a shared assembly table.

The technician who has the third missing component completes the drone assembly and marks the drone as ready for deployment, signalling the agent (assembler) upon completion. The agent then places another two of the three components on the assembly table, and the cycle repeats.

Follow the design process outlined in class to develop a Java monitor that synchronizes the agent (assembler) and the technicians. Produce UML sequence diagrams and UML class diagrams for your design. Then write a program (following your design) to simulate the assembler and the technicians. The program should run until 20 autonomous drones have been assembled and marked ready for deployment.

Work Products

1. A “README.txt” file explaining the names of your files, set up instructions, etc.
2. Your design diagrams including:
   - a UML sequence diagram showing one sample interaction between the Main class (Client), Technician class and Assembler/Agent class.
   - a UML class diagram for your system.
     - o You don't need to show the full diagram for Thread or Runnable. You can just show that you are using them. Refer to lecture slides for examples.

Hand drawn (and then scanned or photographed) diagrams are acceptable, as long as they are neatly drawn and your handwriting is legible.

3. The Java source code for your classes, as well as any files required to run these files in IntelliJ. Your code **must** demonstrate good programming style, be well documented, etc. There are marks for this.

Reminder

The TAs will mark your assignments in the lab environment. It is your responsibility to ensure that your code works in that environment, and that any software required for viewing any text/diagrams is also present in that lab. You must use the IntelliJ IDE.

Submitting Assignments

Assignments are to be submitted electronically using Brightspace. Emailed submissions will not be accepted. See the course outline for the procedure to follow if illness causes you to miss the deadline.

Please see the master schedule on the main course website (PDF), or the Course Schedule for the lab section you are enrolled in (click on “Full Schedule”), for due dates. Assignments are due at 9:00pm on Saturday. Submit early and submit often!