

# CNT 4714 – Project 2 – Fall 2020

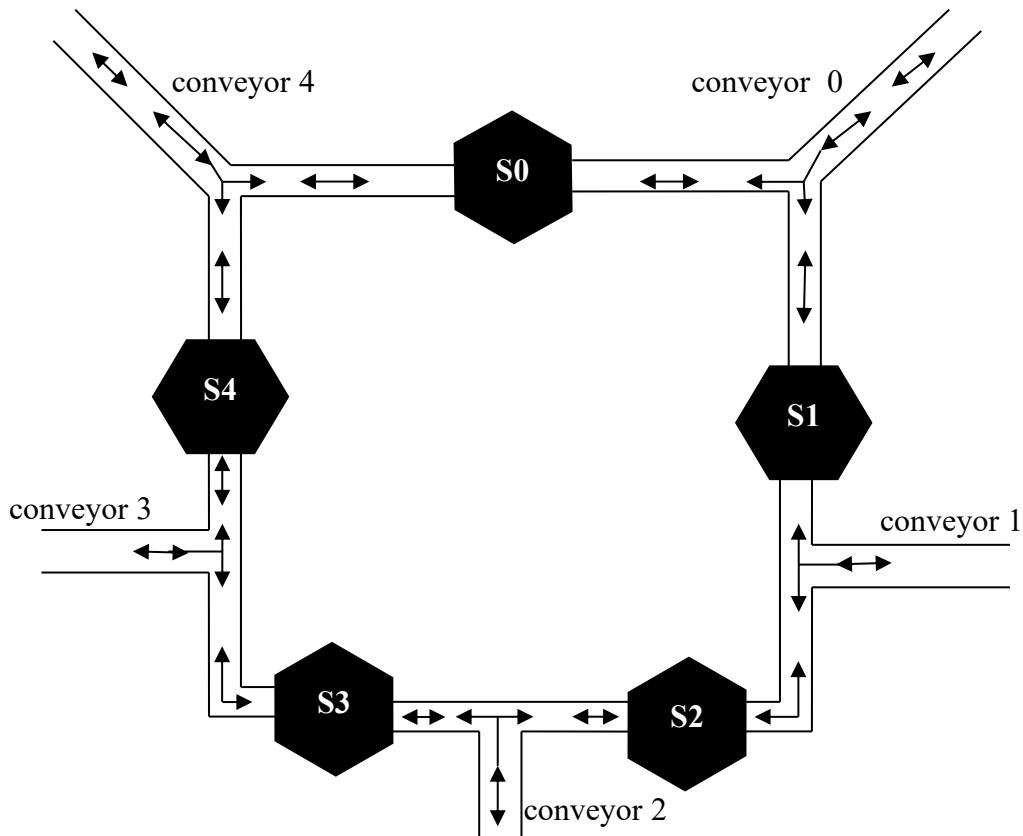
**Title:** “Project 2: Multi-threaded Programming in Java”

**Points:** 100

**Due Date:** Sunday October 4, 2020 by 11:59pm WebCourses time

**Objectives:** To practice programming an application with multiple threads of execution and synchronizing their access to necessary shared objects.

**Description:** In this programming assignment you will simulate the package shipping management system for an automated package shipping operation similar to the one depicted here:



This example package shipping operation has five routing stations (S0 – S4), each of which has an input and output conveyor connecting to conveyor lines (C0 – C4) that go elsewhere in the system. Resources were limited when the system was built so each conveyor going to the rest of the facility must be shared between two routing stations. Since each routing station simultaneously needs an input and output connection to function, access to the shared conveyor lines must be strictly regulated. Flow direction is not important in our simulation.

You have been hired to design a simulator for a new package management system being built with the same design, but possibly fewer/more stations. You are to implement this simulator in Java and have each routing station function in its own thread. A routing station moves packages from one of its connected conveyors to the other. A station’s workload is the number of times that a routing

station needs to have exclusive access to the input and output conveyors during the simulation. Once a routing station is granted access to both conveyors it calls its `doWork()` method during which it will flow packages down each of its connected conveyors (of course it must verify that it has access and isn't in conflict with another routing station). After the packages-in and packages-out methods are run, the workload of the routing station is reduced by 1 and the routing station will release both of the conveyors and signal waiting routing stations that the conveyors are available. After executing a flow and releasing its conveyors, a routing station should sleep for some random period of time. A routing station's thread stops running when its workload reaches 0. To prevent deadlock, ensure that each routing station acquires locks on the conveyors it needs in increasing numerical order.

### Restrictions:

1. Your source files should begin with comments containing the following information:

```
/*
```

```
Name: <your name goes here>
```

```
Course: CNT 4714 Fall 2020
```

```
Assignment title: Project 2 – Multi-threaded programming in Java
```

```
Date: October 4, 2020
```

```
Class: <name of class goes here>
```

```
*/
```

2. **Do not** use a monitor to control the synchronization in your program (i.e., do not use the Java `synchronize` statement).

### Input Specification:

Your program must initially read from a text file (`config.txt`) to gather configuration information for the simulator. The first line of the text file will be the number of routing stations to use during the simulation. Afterwards, there will be one line for each station. These lines will hold the amount of work each station needs to process (i.e, the number of times it needs to move packages down the conveyor system). Only use integers in your configuration file, decimals will not be needed. You can assume that the maximum number of stations will be 10.

### Output Specification:

Your simulator must output the following text to let the user know what the simulator is doing in each of these situations:

1. An input conveyor is set:

```
Routing Station X: input connection is set to conveyor number n
```

2. An output conveyor is set:

```
Routing Station X: output connection is set to conveyor number n
```

3. A stations workload is set:

```
Routing Station X: Workload set. Station X has a total of n package groups to move.
```

4. A station is granted access to its input conveyor:

```
Station X: holds lock on input conveyor n
```

5. A station is granted access to its output conveyor:  
**Station X: holds lock on output conveyor n**
6. A station releasing access to its input conveyor:  
**Station X: unlocks input conveyor n**
7. A station releasing access to its output conveyor:  
**Station X: unable to lock output conveyor - releasing lock on input conveyor n.**
8. A station cannot obtain its second lock (on the output conveyor):  
**Station X: unlocks output conveyor n**
9. A station successfully flows packages on input conveyor:  
**Station X: successfully moves packages into station on input conveyor n.**
10. A station successfully flows packages on output conveyor:  
**Station X: successfully moves packages out of station on output conveyor n.**
11. A station completes a flow:  
**Station X: has n package groups left to move.**
12. A station has completed its workload:  
**\* \* Station X: Workload successfully completed. \* \***

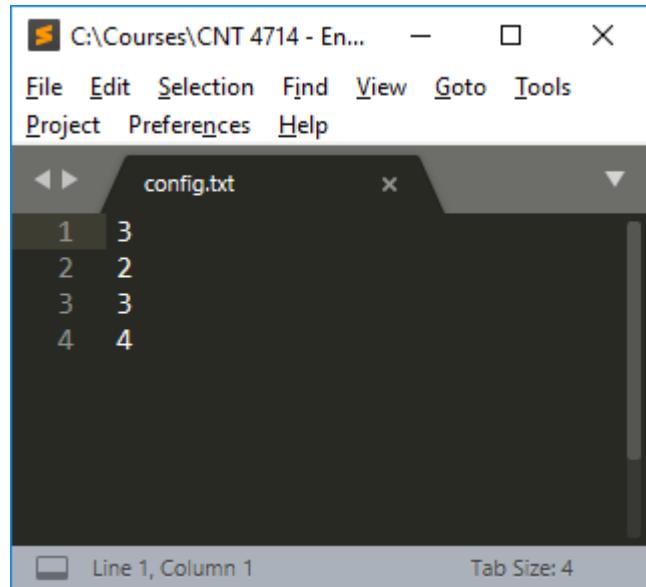
### **Deliverables:**

Submit the following items via WebCourses no later than 11:59pm October 4, 2020.

- (1) All of your .java files.
- (2) A copy of a sample execution of your program, i.e., the output produced by your simulator (this should just be a text file). In your IDE redirect console output to a file, do this and include a copy of the output file produced by your program.

### **Additional Information:**

Actual simulation run in Eclipse (console output redirected in this example) with **config.txt** containing **3 2 3 4**, is shown below.



The screenshot shows a dark-themed Eclipse console window titled "config.txt". The window contains the following text:

```
1 3
2 2
3 3
4 4
```

The status bar at the bottom indicates "Line 1, Column 1" and "Tab Size: 4".

eclipse 2020-06 workspace - CNT 4714 - Project 2 - Fall 2020/src/RoutingStation.java - Eclipse IDE

Problems Javadoc Declaration Console

<terminated> PackageManagementFacility [Java Application] /Library/Java/JavaVirtualMachines/jdk-14.0.2.jdk/Contents/Home/bin/java (Sep 14, 2020, 7:03:13 PM - 7:03:15 PM)  
[Console output redirected to file:/Users/markllewellyn/eclipse 2020-06 workspace/CNT 4714 - Project 2 - Fall 2020/simulation-output.txt]

```
* * * SIMULATION BEGINS * * *

Routing Station 0 Has Total Workload Of 2
Routing Station 1 Has Total Workload Of 3
Routing Station 2 Has Total Workload Of 4

Routing Station 0: input connection is set to conveyor number C0.
Routing Station 2: input connection is set to conveyor number C2.
Routing Station 2: output connection is set to conveyor number C1.
Routing Station 2: Workload set. Station 2 has a total of 4 package groups to move.
Routing Station 2: holds lock on input conveyor C2.
Routing Station 2: holds lock on output conveyor C1.
Routing Station 1: input connection is set to conveyor number C1.
Routing Station 1: output connection is set to conveyor number C0.
Routing Station 1: Workload set. Station 1 has a total of 3 package groups to move.
Routing Station 0: output connection is set to conveyor number C2.
Routing Station 0: Workload set. Station 0 has a total of 2 package groups to move.
Routing Station 0: holds lock on input conveyor C0.
Routing Station 0: unable to lock output conveyor - releasing lock on input conveyor C0.
Routing Station 2: **** Now moving packages. ****
Routing Station 2: successfully moves packages into station on input conveyor 2.
Routing Station 2: successfully moves packages out of station on output conveyor 1.
Routing Station 2: has 3 package groups left to move.

Routing Station 2: unlocks input conveyor C2.
Routing Station 2: unlocks output conveyor C1.
Routing Station 2: holds lock on input conveyor C2.
Routing Station 1: holds lock on input conveyor C1.
Routing Station 2: unable to lock output conveyor - releasing lock on input conveyor C2.
Routing Station 1: holds lock on output conveyor C0.
Routing Station 1: **** Now moving packages. ****
Routing Station 1: successfully moves packages into station on input conveyor 1.
Routing Station 1: successfully moves packages out of station on output conveyor 0.
Routing Station 1: has 2 package groups left to move.
```

eclipse 2020-06 workspace - CNT 4714 - Project 2 - Fall 2020/src/RoutingStation.java - Eclipse IDE

Problems Declaration Console

<terminated> PackageManagementFacility [Java Application] /Library/Java/JavaVirtualMachines/jdk-14.0.2.jdk/Contents/Home/bin/java (Sep 14, 2020, 7:03:13 PM – 7:03:15 PM)

```
Routing Station 2: holds lock on input conveyor C2.  
Routing Station 2: unable to lock output conveyor - releasing lock on input conveyor C2.  
Routing Station 1: unlocks input conveyor C1.  
Routing Station 0: holds lock on input conveyor C0.  
Routing Station 1: unlocks output conveyor C0.  
Routing Station 0: holds lock on output conveyor C2.  
Routing Station 0: **** Now moving packages. ****  
Routing Station 0: successfully moves packages into station on input conveyor 0.  
Routing Station 1: holds lock on input conveyor C1.  
Routing Station 0: successfully moves packages out of station on output conveyor 2.  
Routing Station 0: has 1 package groups left to move.
```

```
Routing Station 1: unable to lock output conveyor - releasing lock on input conveyor C1.  
Routing Station 0: unlocks input conveyor C0.  
Routing Station 0: unlocks output conveyor C2.  
Routing Station 0: holds lock on input conveyor C0.  
Routing Station 0: holds lock on output conveyor C2.  
Routing Station 0: **** Now moving packages. ****  
Routing Station 0: successfully moves packages into station on input conveyor 0.  
Routing Station 0: successfully moves packages out of station on output conveyor 2.  
Routing Station 0: has 0 package groups left to move.
```

```
Routing Station 1: holds lock on input conveyor C1.  
Routing Station 1: unable to lock output conveyor - releasing lock on input conveyor C1.  
* * Station 0: workload successfully completed. * *
```

```
Routing Station 0: unlocks input conveyor C0.  
Routing Station 0: unlocks output conveyor C2.  
Routing Station 2: holds lock on input conveyor C2.  
Routing Station 2: holds lock on output conveyor C1.  
Routing Station 2: **** Now moving packages. ****  
Routing Station 2: successfully moves packages into station on input conveyor 2.  
Routing Station 2: successfully moves packages out of station on output conveyor 1.  
Routing Station 2: has 2 package groups left to move.
```

eclipse 2020-06 workspace - CNT 4714 - Project 2 - Fall 2020/src/RoutingStation.java - Eclipse IDE

Problems Javadoc Declaration Console

<terminated> PackageManagementFacility [Java Application] /Library/Java/JavaVirtualMachines/jdk-14.0.2.jdk/Contents/Home/bin/java (Sep 14, 2020, 7:03:13 PM - 7:03:15 PM)

```
Routing Station 2: unlocks input conveyor C2.  
Routing Station 2: unlocks output conveyor C1.  
Routing Station 2: holds lock on input conveyor C2.  
Routing Station 2: holds lock on output conveyor C1.  
Routing Station 2: **** Now moving packages. ****  
Routing Station 2: successfully moves packages into station on input conveyor 2.  
Routing Station 2: successfully moves packages out of station on output conveyor 1.  
Routing Station 2: has 1 package groups left to move.  
  
Routing Station 2: unlocks input conveyor C2.  
Routing Station 2: unlocks output conveyor C1.  
Routing Station 2: holds lock on input conveyor C2.  
Routing Station 2: unable to lock output conveyor - releasing lock on input conveyor C2.  
Routing Station 1: holds lock on input conveyor C1.  
Routing Station 1: holds lock on output conveyor C0.  
Routing Station 1: **** Now moving packages. ****  
Routing Station 1: successfully moves packages into station on input conveyor 1.  
Routing Station 1: successfully moves packages out of station on output conveyor 0.  
Routing Station 1: has 1 package groups left to move.  
  
Routing Station 1: unlocks input conveyor C1.  
Routing Station 1: unlocks output conveyor C0.  
Routing Station 1: holds lock on input conveyor C1.  
Routing Station 1: holds lock on output conveyor C0.  
Routing Station 1: **** Now moving packages. ****  
Routing Station 1: successfully moves packages into station on input conveyor 1.  
Routing Station 1: successfully moves packages out of station on output conveyor 0.  
Routing Station 1: has 0 package groups left to move.  
  
* * Station 1: workload successfully completed. * *
```

Routing Station 1: unlocks input conveyor C1.  
Routing Station 1: unlocks output conveyor C0.  
Routing Station 2: holds lock on input conveyor C2.  
Routing Station 2: holds lock on output conveyor C1.  
Routing Station 2: \*\*\*\* Now moving packages. \*\*\*\*  
Routing Station 2: successfully moves packages into station on input conveyor 2.  
Routing Station 2: successfully moves packages out of station on output conveyor 1.  
Routing Station 2: has 0 package groups left to move.

\* \* Station 2: workload successfully completed. \* \*

Routing Station 2: unlocks input conveyor C2.  
Routing Station 2: unlocks output conveyor C1.

\* \* \* \* ALL WORKLOADS COMPLETE \* \* \* \* SIMULATION ENDS \* \* \* \*

```
simulation-output.txt UNREGISTERED
config2.txt simulation-output.txt

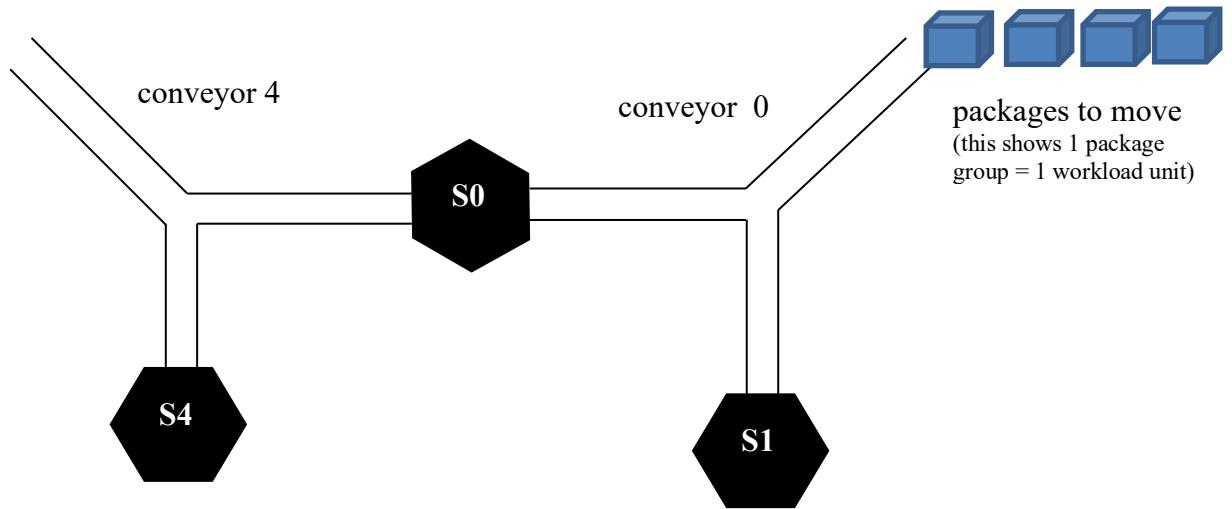
1 | 
2 | * * * SIMULATION BEGINS * * *
3 |
4 Routing Station 0 Has Total Workload Of 2
5 Routing Station 1 Has Total Workload Of 3
6 Routing Station 2 Has Total Workload Of 4
7 |
8 |
9 Routing Station 0: input connection is set to conveyor number C0.
10 Routing Station 2: input connection is set to conveyor number C2.
11 Routing Station 2: output connection is set to conveyor number C1.
12 Routing Station 2: Workload set. Station 2 has a total of 4 package groups to move.
13 Routing Station 2: holds lock on input conveyor C2.
14 Routing Station 2: holds lock on output conveyor C1.
15 Routing Station 1: input connection is set to conveyor number C1.
16 Routing Station 1: output connection is set to conveyor number C0.
17 Routing Station 1: Workload set. Station 1 has a total of 3 package groups to move.
18 Routing Station 0: output connection is set to conveyor number C2.
19 Routing Station 0: Workload set. Station 0 has a total of 2 package groups to move.
20 Routing Station 0: holds lock on input conveyor C0.
21 Routing Station 0: unable to lock output conveyor - releasing lock on input conveyor C0.
22 Routing Station 2: **** Now moving packages. ****
23 Routing Station 2: successfully moves packages into station on input conveyor 2.
24 Routing Station 2: successfully moves packages out of station on output conveyor 1.
25 Routing Station 2: has 3 package groups left to move.
26 |
27 |
28 Routing Station 2: unlocks input conveyor C2.
29 Routing Station 2: unlocks output conveyor C1.
30 Routing Station 2: holds lock on input conveyor C2.
31 Routing Station 1: holds lock on input conveyor C1.
32 Routing Station 2: unable to lock output conveyor - releasing lock on input conveyor C2.
33 Routing Station 1: holds lock on output conveyor C0.
34 Routing Station 1: **** Now moving packages. ****
35 Routing Station 1: successfully moves packages into station on input conveyor 1.
36 Routing Station 1: successfully moves packages out of station on output conveyor 0.
37 Routing Station 1: has 2 package groups left to move.
38 |
39 |
40 Routing Station 2: holds lock on input conveyor C2.
41 Routing Station 2: unable to lock output conveyor - releasing lock on input conveyor C2.
42 Routing Station 1: unlocks input conveyor C1.
43 Routing Station 0: holds lock on input conveyor C0.
44 Routing Station 1: unlocks output conveyor C0.
45 Routing Station 0: holds lock on output conveyor C2.
46 Routing Station 0: **** Now moving packages. ****
47 Routing Station 0: successfully moves packages into station on input conveyor 0.
48 Routing Station 1: holds lock on input conveyor C1.
```

First part of the simulation output as redirected from the console to an output file and viewed via Sublime.

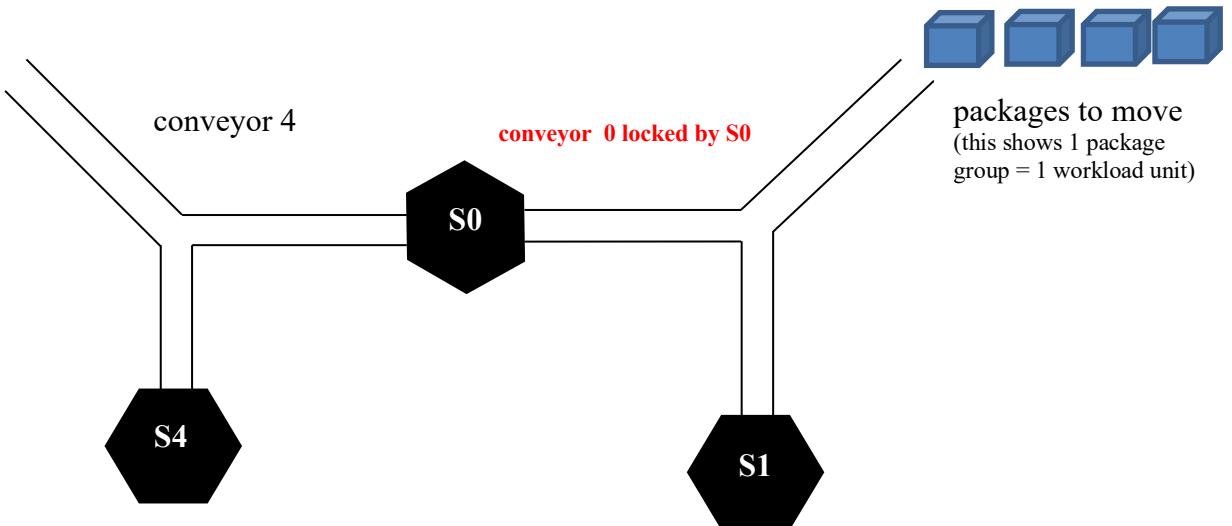
The example scenario below may help you to understand what is happening with a routing station and the acquisition of locks and work flow. Illustration is based only on S0 from the original configuration.

### **Illustration of the timeline of a station in execution**

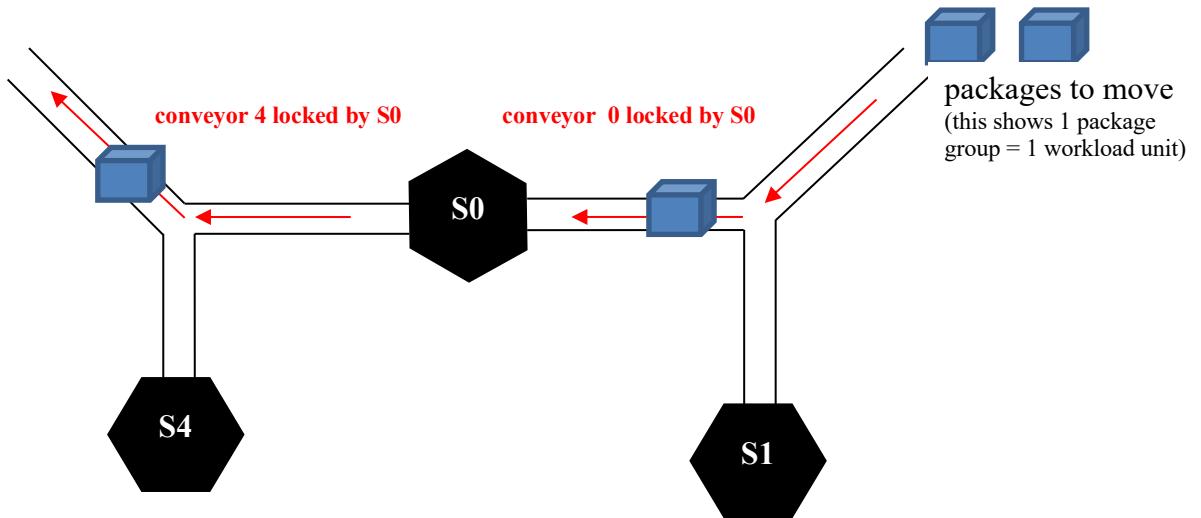
**Time 1:** Station 0 needs to move a package group from input conveyor (conveyor 0) to the output conveyor (conveyor 4). It must acquire the locks on conveyor 0 and conveyor 4, in that order. If Station 1, currently holds the lock on conveyor 0, then Station 0 is blocked and must wait for conveyor 0 to become available. Note that Station 0 cannot attempt to lock conveyor 4 if it is unable to obtain the lock on conveyor 0.



**Time 2:** Station 0 acquires lock on conveyor 0. Attempts to lock conveyor 4. If conveyor 4 is not available (Station 4 currently owns lock on conveyor 4), then Station 0 must release lock on conveyor 0 and block until both conveyor 0 and conveyor 4 can be obtained.



**Time 3:** Station 0 has acquired locks on conveyor 0 and conveyor 4. Packages now moving.



**Time 4:** Station 0 has successfully moved 1 workload unit of packages from input side to output side. Station 0 releases locks on conveyor 0 and conveyor 4. Station 0 goes idle for random period of time if its workload is not yet 0, otherwise, Station 0 terminates.

