

Problem Set 7

Due: 4:30PM, Friday April 12, 2002

Problem 1. Robots in a manufacturing facility

Background

You are designing a materials management system for a stereo manufacturing facility.

In this facility there are 2 complete assembly lines that build and crate stereos. The stereos are then placed on powered conveyors and fed into a robot's pick stations. The stereos arrive in a queue at the end of the conveyor and are picked, one by one, by the robot and placed on its carrier.

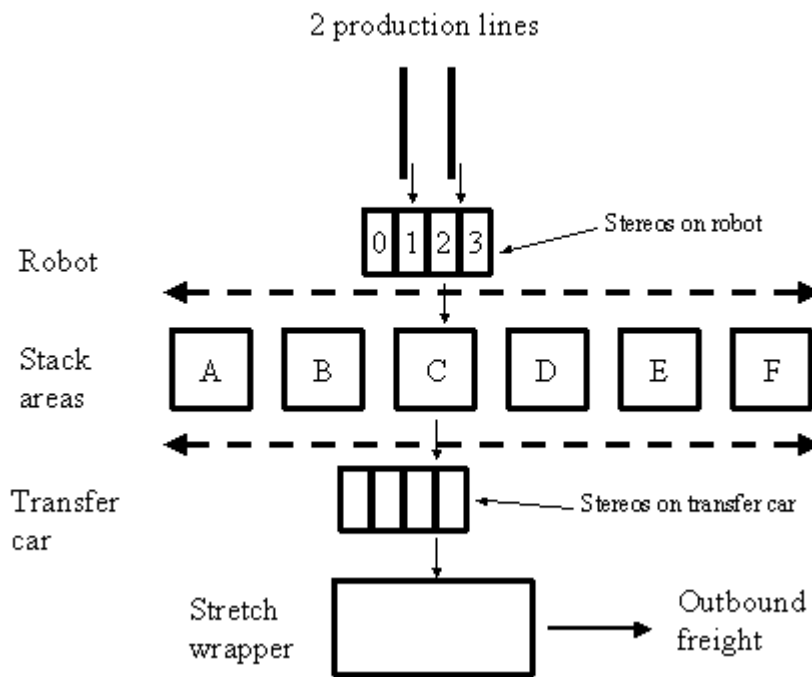
The figure below shows an example of a robot picking up a stereo from the end of a conveyor. It takes them in order (first in, first loaded) from the conveyor. (In this homework, we ignore the details of exactly how the robot places them on its carrier.)



The vacuum head of the robot is brought to the pick location and the product is "engaged" by the vacuum head and placed on the carrier. Sensors verify that the product remains engaged by the vacuum head during the pick move. After 4 stereos are loaded on the carrier, the robot carrier then moves to the stacking area or place station. At the place stations, stacks of products are being erected. The product is placed in its assigned stack location. Once the stack move is completed, the robot is free to service pick requests from other assembly lines' pick stations.

When an order is placed for stereos to be shipped to a distributor, this results in a transfer request from the stacking area to the stretch wrap station. When a transfer is requested, the transfer car (another robot) is driven to the stacking area (place station). Transfer chain conveyors on the transfer car and respective place station are turned on and the product stack moves from the place station to the transfer car. After the product stack is on the car, the car drives on rails to the in-feed of an automated stretch wrapper where the transfer car is unloaded in a similar fashion. The transfer car is now available to service other transfer requests. The stretch wrapper wraps the product stack and presents it for removal by lift truck operators to load outbound trucks or containers.

The figure below summarizes the operation of the materials management system. The problem statement discusses the details on this figure.



This is similar to actual production systems in use, which are generally controlled by a PC with a GUI for an operator to oversee the entire process. A single operator can manage all the conveyor belts, pick stations, place stations, transfer car and stretch wrap station. These systems have a 'teach mode' to allow the operator to make the desired movements manually with the robot(s) and record them; this is more effective than programming. The movements can then be edited and movements taught for one station can be run on all stations.

Problem statement

Your program will simulate the operation of the stereo plant. You must model the following processes:

1. The plant makes 6 types of stereos. You may number them 0-5.
2. Each stereo has a type and a serial number when it is produced.
3. The plant has two production lines. You should model these as queues.
 - a. Line 0 makes 8 stereos per minute.
 - b. Line 1 makes 4 stereos per minute.

4. The user is prompted, at the start of each minute, to enter the stereo type (0-5) to be produced on line 0 and line 1. After the products are entered, they are 'produced' and added to the queues (production lines).
5. The production lines assign a unique serial number to each stereo. Serial numbers start with 0 and increase by one for each stereo built. First, line 0 obtains serial numbers for all stereos it will build in the next minute, and then line 1 obtains serial numbers for the stereos it will build in the next minute. For simplicity, you may assume that line 0 completes its production for the minute before line 1 begins its production.
6. The plant has one robot that moves stereos in groups of 4 from the end of the production lines to the stacking area. It removes them from the queue at the end of the production line, and places them in a stacking area, described below. Since Line 0 makes 8 stereos in a minute, and the robot can only carry 4 stereos at a time, the robot must make 2 trips to service line 0 during each minute.
7. The plant has 6 stacking areas, one for each stereo type. Each stacking area consists of 4 physical stacks where stereos are placed. The heights of the stacks must be kept as even as possible.
8. Customers may place an order for any quantity of each stereo type.
 - a. We handle one customer order per minute, at the 'end' of the production minute after the stereos have been placed in the stacking areas.
 - b. The user is prompted to enter the quantity of each stereo type (0-5) in the order.
 - c. In this homework, you may assume that there will be sufficient stereos to fill the order. You don't need to handle the case where more stereos of a type are requested than available. When an order is made, the transfer car is activated to fill it.
9. The plant has one transfer car that moves stereos in any quantity (a simplifying assumption, of course) from the stacking area to the stretch wrap station. The transfer car moves stereos in a single order to the stretch wrap station. It must keep the height of the stacks as even as possible in the stacking areas.
10. When the transfer car makes a movement it outputs (using System.out.println):
 - a. Order number, starting at 1 and incrementing by 1
 - b. For each stereo in the order:
 - i. The stereo type (0-5)
 - ii. The stereo serial number

11. You don't need to model the stretch wrap station or outbound freight. Once the transfer car has output the order and its details, you have completed that minute's activities.
12. Your program should model 5 minutes of activity in the plant. (5 complete passes through the preceding steps. You do not need to account for actual clock time in your program).

Your program should use data structures (stacks and queues) appropriately to model the physical movement of the speakers through the plant.

1. You should model a production line as a Queue. Use the ArrayQueue or other implementation of the Queue interface presented in lecture.
2. You should create a StackingArea class to hide the implementation details of the stack management. You should model the stacking areas as arrays of Stacks. Use the ArrayStack or other implementation of the Stack interface presented in lecture. You will need to extend it slightly to be able to manage the height of the stacks.
3. You should model the Robot and TransferCar as classes with appropriate behaviors. They can be quite simple.
4. You should create a Product class to represent the objects (stereos) being produced (added to the queues) and managed in the stacks.
5. The main() method can contain the overall logic to be executed:
 - a. For each of the 5 minutes in the plant operation, prompt for the products to be produced on each line, invoke the correct operations in the production lines and robot, prompt for the order quantity of each speaker, and invoke the correct operations in the transfer car.
6. Your program does not need to be general. It only needs to operate with 2 production lines, 6 products, etc. You may use fixed-dimension arrays where convenient, etc.

Be careful about name conflicts. Java has Stack, Queue and Robot classes, among others.

Extra Credit

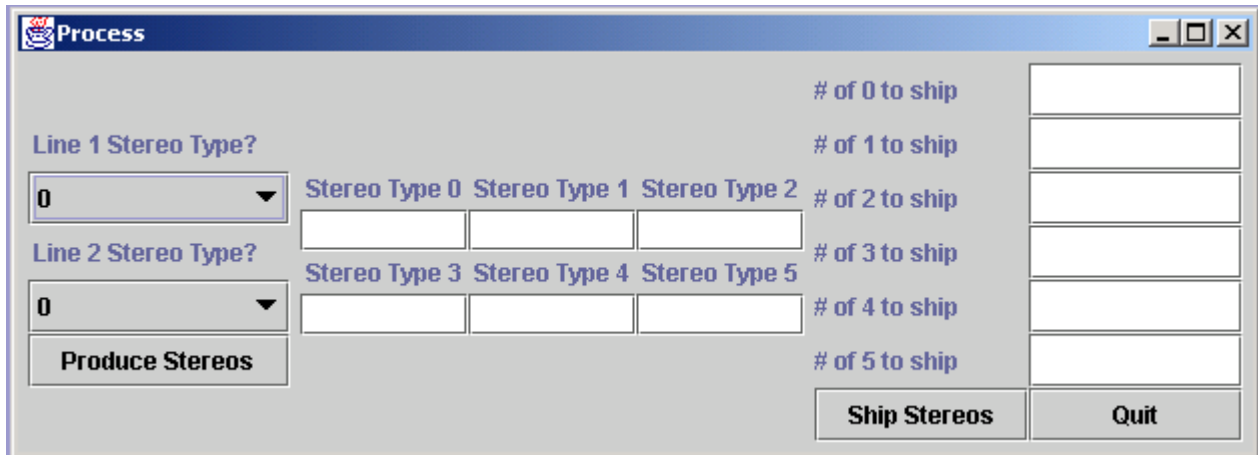
You may write a graphical user interface (GUI) for the primary program in the problem set for a maximum of 20 extra credit points. This is an option in problem sets 6 through 10; you may get extra credit for building a GUI for one (only) of these problem sets. In general, you are free to design the GUI as you wish; this is a 'blank piece of paper' exercise. You may not use System.out.println in your solution; all input and output must be done using Swing. You should hand in two solutions if you create a GUI:

- Hand in the regular assignment, without a GUI, as described in the assignment. This allows us to grade the main part of the assignment without having to worry about any potential errors introduced by your GUI.

- Hand in the full solution using the GUI. This second submission will be graded based only on the GUI, for a maximum of 20 points. Only the GUI (and its immediate interfaces to the rest of the code) will be graded. You will receive between 0 and 20 points; even receiving 0 points cannot hurt or lower your grade on the homework overall.

Specific requirements for Problem Set 7: Your problem must display elements (production line, stacking area, transfer car, stretch wrap station) of the plant and of the user input

graphically to earn full credit. The representation can be stylized or abstract, but it must be graphical in some sense. Here is a possible rendition of the GUI:



Turnin

Turnin Requirements

- Problem 1: Email only. No hardcopy required.
- Problems 2 & 3: Hardcopy and electronic copy of ALL source code (all .java files).
- Place a comment with your name, username, section, TA's name, assignment number, and list of people with whom you have discussed the problem set on ALL files you submit.
- DO NOT turn in electronic or hardcopies of compiled byte code (.class files).

Electronic Turnin

To electronically turn in your problem sets, run Netscape

Then go to the 1.00 web page at:

<http://command.mit.edu/1.00Spring02>

Click on the "Submit Assignment" button. Be sure to set the Selection Bar to Problem Set 1 or your files may be lost. Finally, go back to the home page and click on the "View" section and be sure that your files were received. If you submit a file twice, the latest version will be graded.

Penalties

- Missing Hardcopy: -10% off problem score if missing hardcopy.
- Missing Electronic Copy: -30% off problem score if missing electronic copy.
- Late Turnin: -20% off problem score if 1 day late. More than 1 day late = NO CREDIT