Sabancı University

Faculty of Engineering and Natural Sciences
CS204 Advanced Programming
Spring 2021

Homework 7 – Simulation of Factory Production Lines using Threads

Due: 26/05/2021, Wednesday, 21:00

PLEASE NOTE:

Your program should be a robust one such that you have to consider all relevant programmer mistakes and extreme cases; you are expected to take actions accordingly!

You can NOT collaborate with your friends and discuss solutions. You have to write down the code on your own. Plagiarism will not be tolerated!

Introduction

In this homework, you are asked to write a **multithreaded** C++ program that simulates two production lines in a factory. One of the production lines is for filling the empty boxes and the other one is for packaging the boxes. Firstly, producer (thread) produces empty boxes and put them in a filling queue. In the filling production line, two different filler workers (threads) always check that queue and if there is an empty box, an available filler worker starts to fill it. After a worker finishes the filling a box, s/he puts it into another queue (packaging queue) to be packaged. In addition to the filler workers, there also are two other workers work on packaging line (packager worker threads) and they check packaging queue all the time if it is empty or not. If it is not empty, one of available packager workers takes a box from that queue and starts to pack it. This process continues until all the boxes are filled and packaged. In this homework, you will simulate these two production lines, two filler and two packager workers, and the producer via dynamic queue data structure and multithreading techniques.

The time for filling and packaging the boxes are probabilistic. Moreover, the time for the producer to produce an empty box is also probabilistic. The parameters of these random values will be input (see Section "Details of Simulation" for details).

In the scope of this homework, simulation means to employ two different queues to model the factory production lines. One of the queues, filling queue, is for the empty boxes that are to be enqueued by the producer thread, and dequeued by the filler worker threads at some random intervals. The other queue, packaging queue, is for the filled boxes that are to be enqueued by the filler worker threads and dequeued by the packager worker threads at some random intervals. Simulation starts after taking the inputs from the keyboard and continues until all of the boxes are filled and packaged. During the simulation, you are going to display some verbose output about the actions of producer and workers (see Section "Details of Simulation" for details).

Using Threads

There will be five threads (other than the main thread) in your program. One thread is for the producer. Two threads are for two workers in the filling line, one for each. The other two threads are for two workers in the packaging line, again one for each.

In the producer thread, the producer will produce boxes one by one at random intervals and enqueue them to the filling queue. Then, the filler workers dequeue the boxes from the filling queue and starts the filling process, which also takes a random duration. After a worker finishes the filling phase for a box, s/he enqueues that filled box into the packaging queue. Meanwhile, as the packaging queue gets populated, packager workers dequeue the filled boxes from the packaging queue and start packaging, which also takes a random duration. Here, please remark that the threads working on the same queue are in race condition as detailed in the next paragraph.

Please use the `HW7DynIntQueue` class given in this homework package as the queue data structure. Generate two such queues, one for filling queue and another one for packaging queue. Moreover, do not forget that the producer and the two filler worker threads contend to operate on filling queue; in other words, the producer thread enqueues and the filler worker threads dequeue on the same filling queue. Similarly, the filler worker and the packager worker threads contend to operate on the packaging queue, which means filler workers enqueue, while the packager workers dequeue. These issues may cause some special cases to deal with in a multithreaded application. For example, there might be a synchronization conflict during enqueuing and dequeuing boxes to/from the queues. Some other special cases and conflicts may occur as well; therefore, you have to take some precautions in the threads to avoid such situations. Please refer to lecture and lab materials, which include methods for dealing with these cases using *mutex*.

At the end, your threads must be joined properly and your program must terminate without any complications/crashes.

Details of Simulation

Before the simulation begins, some inputs (total 7 of them) are entered via keyboard. First, *total_box_count* is entered. *total_box_count* is the total number of boxes which are going to be produced by the producer during the simulation. In other words, the simulation will finish when *total_box_count* boxes are completely prepared (filled and packaged). Moreover, you should not allow more than *total_box_count* boxes to be created. Thus, at the end of the simulation, all boxes should be processed by the workers and the queues must be empty.

Then, the parameters of the random duration for a box creation, random duration for filling a box and random duration for packaging a box must be entered via keyboard. The duration before creation of a box (before the first box and before all other boxes) is a random value in seconds. This random value is between two integer parameters: *min_production* and *max_production* (*min* and *max* values are included in the range). Before enqueuing each box, a different random number is picked within this range. The filling and packaging durations are also determined similarly, but their parameters are different; for filling a box, it is a random integer value between and including *min_filling_time* and *max_filling_time*; for packaging a box, it is a random integer value between and including *min_packaging_time* and *max_packaging_time*. Of course, the random packaging and filling time will be picked for each box separately. You do not need to make any input checks for these parameters. You can assume that *min* values are always less than or equal to the corresponding *max* values and *min* values are greater than or equal to 1 second. To generate such random numbers, **do not use the `RandGen` class from** CS201 in multithreaded applications because it is not thread-safe. Instead, **you must use the following thread-safe function, `random_range`, for generating random waiting times inside the threads**. This function returns a random number between (and including) `min` and `max` parameters. In the threads, you can call it by passing minimum and maximum waiting times of the corresponding thread as arguments.

```
#include <random>
#include <time.h>

int random_range(const int & min, const int & max) {
    static mt19937 generator(time(0));
    uniform_int_distribution<int> distribution(min, max);
    return distribution(generator);
}
```

Simulation starts when the user enters all the inputs through keyboard. During the simulation, boxes will be created by producer, and the workers will fill and package boxes in corresponding threads as explained before.

Each new box must be associated with a consecutive ID number starting with 1. You may simulate creation, filling and packaging time of a box by sleeping the threads using the `this_thread::sleep_for(chrono::seconds(`*time_in_seconds*`))` command of `thread` (for `this_thread::sleep_for`) and `chrono` (for `chrono::seconds`) libraries.

At the beginning of the program (after getting the inputs and just before the beginning of the simulation), you have to display a message saying that the simulation is starting at the current time (please refer to the lecture notes to see how the current system time is displayed). During the simulation, producer thread should display the creation details of each box. These details contain the ID of the box, the filling queue size after enqueue operation and the time of creation. Similarly, each worker thread should display the details of the filling and packaging process. At the beginning and at the end of a process, the tag of the workers (filler worker 1, filler worker 2, packager worker 1, or packager worker 2), the ID of the box and the time of the process should be displayed. Additionally, at the beginning of the process, the size of the corresponding queue (after dequeuing) should be displayed as well. At the end of the simulation, a message saying that the simulation is ended must be displayed together with the current time. Please see the sample run section for some examples.

Two important rules about the simulation details and use of threads:

- Here please remark that a single line of output from a particular thread may be interleaved by the output of another thread if you do not take appropriate precautions. If this happens, the outputs mix up and become messy. You have to code appropriately, with the help of a dedicated mutex, in order not to end up with such an undesirable situation. Here by "dedicated" we mean this; the mutex that you use for tidy output should not be the ones that you use for accessing the shared queue objects. These three critical sections are different concepts and must be handled via three different mutexes.
- Do not sleep a thread while a mutex is locked. We use sleeping to simulate the creation/filling/packaging duration. We do not access shared resources while doing so. Thus, sleeping a thread while a mutex is locked is totally nonsense.
- Any violation to these rules will be penalized severely (50 points or more).

`HW7DynIntQueue` class, use of global variables, number of parameters for thread constructor

We provide a header file (HW7DynIntQueue.h) and its implementation (HW7DynIntQueue.cpp) together with this homework. **You have to use this queue class without any change**. In order to use it, include both the header and the .cpp to your project. Of course, you have to copy these files to appropriate folders in your project.

Finally, a good news; you may use global variables in this homework. Actually, it would be miserable not to use them in a program that has several threads. However, we kindly request you not to

exaggerate the global usage since after a certain point you may lose control over your program (as the famous Turkish proverb says "azı karar, coğu zarar").

Some versions of C++ compilers impose a limitation for the number of parameters for the `thread` constructor. It is 5 (except the entry point function address) for VC++ 2012 and if you exceed this, you receive compilation errors. If this amount of parameters is not sufficient for you, you may make some of them global, or pack them into structs/arrays, or increase this limit using the steps explained below (taken from stackoverflow.com).

> Right click on your project in Solution Explorer. Select Properties... and naviagete to C/C++ -> Preprocessor and set add _VARIADIC_MAX=10 to "Preprocessor Definitions". Make sure you do that for all the configurations of the project (debug, release, etc.) – JP Flouret Nov 13 '14 at 22:44

Sample Runs

Some sample runs are given below, but these are not comprehensive, therefore you have to consider all cases, to get full mark.

Due to the probabilistic nature of the homework and due to scheduling of threads, same inputs may yield different outputs for your code. However, the order of the events must be consistent with the homework requirements and the given inputs. Nevertheless, occasional (i.e. rare) inconsistencies in the display order of the events occurred at the same time are acceptable.

The inputs from the keyboard are written in ***`boldface and italic`***.

Sample Run 1:

```
Please enter the total number of items:6
Please enter the min-max waiting time range of producer:
Min: 1
Max: 1
Please enter the min-max waiting time range of filler workers:
Min: 1
Max: 1
Please enter the min-max waiting time range of packager workers:
Min: 1
Max: 1
Simulation starts 17:56:46
Producer has enqueued a new box 1 to filling queue (filling queue size is 1): 17:56:47
Filler 1 started filling the box 1 (filling queue size is 0): 17:56:47
Filler 1 finished filling the box 1: 17:56:48
Producer has enqueued a new box 2 to filling queue (filling queue size is 1): 17:56:48
Filler 2 started filling the box 2 (filling queue size is 0): 17:56:48
Filler 1 put box 1 into packaging queue (packaging queue size is 1): 17:56:48
Packager 2 started packaging the box 1 (packaging queue size is 0): 17:56:48
Filler 2 finished filling the box 2: 17:56:49
Packager 2 finished packaging the box 1: 17:56:49
Producer has enqueued a new box 3 to filling queue (filling queue size is 1): 17:56:49
Filler 1 started filling the box 3 (filling queue size is 0): 17:56:49
Filler 2 put box 2 into packaging queue (packaging queue size is 1): 17:56:49
Packager 1 started packaging the box 2 (packaging queue size is 0): 17:56:49
Filler 1 finished filling the box 3: 17:56:50
Packager 1 finished packaging the box 2: 17:56:50
Producer has enqueued a new box 4 to filling queue (filling queue size is 1): 17:56:50
Filler 2 started filling the box 4 (filling queue size is 0): 17:56:50
Filler 1 put box 3 into packaging queue (packaging queue size is 1): 17:56:50
Packager 2 started packaging the box 3 (packaging queue size is 0): 17:56:50
Filler 2 finished filling the box 4: 17:56:51
Producer has enqueued a new box 5 to filling queue (filling queue size is 1): 17:56:51
Filler 1 started filling the box 5 (filling queue size is 0): 17:56:51
Filler 2 put box 4 into packaging queue (packaging queue size is 1): 17:56:51
```

```
Packager 1 started packaging the box 4 (packaging queue size is 0): 17:56:51
Packager 2 finished packaging the box 3: 17:56:51
Producer has enqueued a new box 6 to filling queue (filling queue size is 1): 17:56:52
Filler 2 started filling the box 6 (filling queue size is 0): 17:56:52
Filler 1 finished filling the box 5: 17:56:52
Packager 1 finished packaging the box 4: 17:56:52
Filler 1 put box 5 into packaging queue (packaging queue size is 1): 17:56:52
Packager 2 started packaging the box 5 (packaging queue size is 0): 17:56:52
Filler 2 finished filling the box 6: 17:56:53
Filler 2 put box 6 into packaging queue (packaging queue size is 1): 17:56:53
Packager 1 started packaging the box 6 (packaging queue size is 0): 17:56:53
Packager 2 finished packaging the box 5: 17:56:53
Packager 1 finished packaging the box 6: 17:56:54
End of the simulation ends: 17:56:54
```

Sample Run 2:

```
Please enter the total number of items:5
Please enter the min-max waiting time range of producer:
Min: 5
Max: 8
Please enter the min-max waiting time range of filler workers:
Min: 2
Max: 3
Please enter the min-max waiting time range of packager workers:
Min: 3
Max: 4
Simulation starts 18:03:40
Producer has enqueued a new box 1 to filling queue (filling queue size is 1): 18:03:46
Filler 1 started filling the box 1 (filling queue size is 0): 18:03:46
Filler 1 finished filling the box 1: 18:03:48
Filler 1 put box 1 into packaging queue (packaging queue size is 1): 18:03:48
Packager 2 started packaging the box 1 (packaging queue size is 0): 18:03:48
Packager 2 finished packaging the box 1: 18:03:51
Producer has enqueued a new box 2 to filling queue (filling queue size is 1): 18:03:54
Filler 2 started filling the box 2 (filling queue size is 0): 18:03:54
Filler 2 finished filling the box 2: 18:03:56
Filler 2 put box 2 into packaging queue (packaging queue size is 1): 18:03:56
Packager 2 started packaging the box 2 (packaging queue size is 0): 18:03:56
Producer has enqueued a new box 3 to filling queue (filling queue size is 1): 18:03:59
Filler 2 started filling the box 3 (filling queue size is 0): 18:03:59
Packager 2 finished packaging the box 2: 18:03:59
Filler 2 finished filling the box 3: 18:04:01
Filler 2 put box 3 into packaging queue (packaging queue size is 1): 18:04:01
Packager 2 started packaging the box 3 (packaging queue size is 0): 18:04:01
Producer has enqueued a new box 4 to filling queue (filling queue size is 1): 18:04:04
Filler 2 started filling the box 4 (filling queue size is 0): 18:04:04
Packager 2 finished packaging the box 3: 18:04:04
Filler 2 finished filling the box 4: 18:04:06
Filler 2 put box 4 into packaging queue (packaging queue size is 1): 18:04:06
Packager 1 started packaging the box 4 (packaging queue size is 0): 18:04:06
Packager 1 finished packaging the box 4: 18:04:09
Producer has enqueued a new box 5 to filling queue (filling queue size is 1): 18:04:11
Filler 2 started filling the box 5 (filling queue size is 0): 18:04:11
Filler 2 finished filling the box 5: 18:04:13
Filler 2 put box 5 into packaging queue (packaging queue size is 1): 18:04:13
Packager 1 started packaging the box 5 (packaging queue size is 0): 18:04:13
Packager 1 finished packaging the box 5: 18:04:16
End of the simulation ends: 18:04:16
```

Sample Run 3:

```
Please enter the total number of items:8
Please enter the min-max waiting time range of producer:
Min: 1
Max: 1
Please enter the min-max waiting time range of filler workers:
Min: 5
Max: 6
```

```
Please enter the min-max waiting time range of packager workers:
Min: 2
Max: 2
Simulation starts 18:08:00
Producer has enqueued a new box 1 to filling queue (filling queue size is 1): 18:08:01
Filler 1 started filling the box 1 (filling queue size is 0): 18:08:01
Producer has enqueued a new box 2 to filling queue (filling queue size is 1): 18:08:02
Filler 2 started filling the box 2 (filling queue size is 0): 18:08:02
Producer has enqueued a new box 3 to filling queue (filling queue size is 1): 18:08:03
Producer has enqueued a new box 4 to filling queue (filling queue size is 2): 18:08:04
Producer has enqueued a new box 5 to filling queue (filling queue size is 3): 18:08:05
Filler 1 finished filling the box 1: 18:08:06
Filler 1 put box 1 into packaging queue (packaging queue size is 1): 18:08:06
Packager 1 started packaging the box 1 (packaging queue size is 0): 18:08:06
Filler 1 started filling the box 3 (filling queue size is 2): 18:08:06
Producer has enqueued a new box 6 to filling queue (filling queue size is 3): 18:08:06
Producer has enqueued a new box 7 to filling queue (filling queue size is 4): 18:08:07
Packager 1 finished packaging the box 1: 18:08:08
Filler 2 finished filling the box 2: 18:08:08
Filler 2 put box 2 into packaging queue (packaging queue size is 1): 18:08:08
Packager 1 started packaging the box 2 (packaging queue size is 0): 18:08:08
Filler 2 started filling the box 4 (filling queue size is 3): 18:08:08
Producer has enqueued a new box 8 to filling queue (filling queue size is 4): 18:08:08
Packager 1 finished packaging the box 2: 18:08:10
Filler 1 finished filling the box 3: 18:08:11
Filler 1 put box 3 into packaging queue (packaging queue size is 1): 18:08:11
Packager 2 started packaging the box 3 (packaging queue size is 0): 18:08:11
Filler 1 started filling the box 5 (filling queue size is 3): 18:08:11
Packager 2 finished packaging the box 3: 18:08:13
Filler 2 finished filling the box 4: 18:08:13
Filler 2 put box 4 into packaging queue (packaging queue size is 1): 18:08:13
Packager 1 started packaging the box 4 (packaging queue size is 0): 18:08:13
Filler 2 started filling the box 6 (filling queue size is 2): 18:08:13
Packager 1 finished packaging the box 4: 18:08:15
Filler 1 finished filling the box 5: 18:08:17
Filler 1 put box 5 into packaging queue (packaging queue size is 1): 18:08:17
Packager 1 started packaging the box 5 (packaging queue size is 0): 18:08:17
Filler 1 started filling the box 7 (filling queue size is 1): 18:08:17
Filler 2 finished filling the box 6: 18:08:18
Filler 2 put box 6 into packaging queue (packaging queue size is 1): 18:08:18
Packager 2 started packaging the box 6 (packaging queue size is 0): 18:08:18
Filler 2 started filling the box 8 (filling queue size is 0): 18:08:18
Packager 1 finished packaging the box 5: 18:08:19
Packager 2 finished packaging the box 6: 18:08:20
Filler 1 finished filling the box 7: 18:08:23
Filler 1 put box 7 into packaging queue (packaging queue size is 1): 18:08:23
Packager 2 started packaging the box 7 (packaging queue size is 0): 18:08:23
Filler 2 finished filling the box 8: 18:08:23
Filler 2 put box 8 into packaging queue (packaging queue size is 1): 18:08:23
Packager 1 started packaging the box 8 (packaging queue size is 0): 18:08:23
Packager 2 finished packaging the box 7: 18:08:25
Packager 1 finished packaging the box 8: 18:08:25
End of the simulation ends: 18:08:25
```

Sample Run 4:

```
Please enter the total number of items:8
Please enter the min-max waiting time range of producer:
Min: 1
Max: 1
Please enter the min-max waiting time range of filler workers:
Min: 1
Max: 1
Please enter the min-max waiting time range of packager workers:
Min: 5
Max: 6
Simulation starts 18:13:43
Producer has enqueued a new box 1 to filling queue (filling queue size is 1): 18:13:44
Filler 2 started filling the box 1 (filling queue size is 0): 18:13:45
Producer has enqueued a new box 2 to filling queue (filling queue size is 1): 18:13:46
Filler 1 started filling the box 2 (filling queue size is 0): 18:13:46
```

```
Filler 2 finished filling the box 1: 18:13:46
Filler 2 put box 1 into packaging queue (packaging queue size is 1): 18:13:46
Packager 2 started packaging the box 1 (packaging queue size is 0): 18:13:46
Filler 1 finished filling the box 2: 18:13:47
Producer has enqueued a new box 3 to filling queue (filling queue size is 1): 18:13:47
Filler 2 started filling the box 3 (filling queue size is 0): 18:13:47
Filler 1 put box 2 into packaging queue (packaging queue size is 1): 18:13:47
Packager 1 started packaging the box 2 (packaging queue size is 0): 18:13:47
Producer has enqueued a new box 4 to filling queue (filling queue size is 1): 18:13:48
Filler 1 started filling the box 4 (filling queue size is 0): 18:13:48
Filler 2 finished filling the box 3: 18:13:48
Filler 2 put box 3 into packaging queue (packaging queue size is 1): 18:13:48
Filler 1 finished filling the box 4: 18:13:49
Producer has enqueued a new box 5 to filling queue (filling queue size is 1): 18:13:49
Filler 2 started filling the box 5 (filling queue size is 0): 18:13:49
Filler 1 put box 4 into packaging queue (packaging queue size is 2): 18:13:49
Producer has enqueued a new box 6 to filling queue (filling queue size is 1): 18:13:50
Filler 1 started filling the box 6 (filling queue size is 0): 18:13:50
Filler 2 finished filling the box 5: 18:13:50
Filler 2 put box 5 into packaging queue (packaging queue size is 3): 18:13:50
Producer has enqueued a new box 7 to filling queue (filling queue size is 1): 18:13:51
Filler 2 started filling the box 7 (filling queue size is 0): 18:13:51
Filler 1 finished filling the box 6: 18:13:51
Filler 1 put box 6 into packaging queue (packaging queue size is 4): 18:13:51
Packager 2 finished packaging the box 1: 18:13:52
Packager 2 started packaging the box 3 (packaging queue size is 3): 18:13:52
Producer has enqueued a new box 8 to filling queue (filling queue size is 1): 18:13:52
Filler 1 started filling the box 8 (filling queue size is 0): 18:13:52
Filler 2 finished filling the box 7: 18:13:52
Filler 2 put box 7 into packaging queue (packaging queue size is 4): 18:13:52
Packager 1 finished packaging the box 2: 18:13:53
Packager 1 started packaging the box 4 (packaging queue size is 3): 18:13:53
Filler 1 finished filling the box 8: 18:13:53
Filler 1 put box 8 into packaging queue (packaging queue size is 4): 18:13:53
Packager 2 finished packaging the box 3: 18:13:57
Packager 2 started packaging the box 5 (packaging queue size is 3): 18:13:57
Packager 1 finished packaging the box 4: 18:13:58
Packager 1 started packaging the box 6 (packaging queue size is 2): 18:13:58
Packager 2 finished packaging the box 5: 18:14:03
Packager 2 started packaging the box 7 (packaging queue size is 1): 18:14:03
Packager 1 finished packaging the box 6: 18:14:03
Packager 1 started packaging the box 8 (packaging queue size is 0): 18:14:03
Packager 2 finished packaging the box 7: 18:14:08
Packager 1 finished packaging the box 8: 18:14:09
End of the simulation ends: 18:14:09
```

Sample Run 5:

```
Please enter the total number of items:7
Please enter the min-max waiting time range of producer:
Min: 1
Max: 1
Please enter the min-max waiting time range of filler workers:
Min: 4
Max: 7
Please enter the min-max waiting time range of packager workers:
Min: 10
Max: 13
Simulation starts 18:18:08
Producer has enqueued a new box 1 to filling queue (filling queue size is 1): 18:18:09
Filler 2 started filling the box 1 (filling queue size is 0): 18:18:09
Producer has enqueued a new box 2 to filling queue (filling queue size is 1): 18:18:10
Filler 1 started filling the box 2 (filling queue size is 0): 18:18:10
Producer has enqueued a new box 3 to filling queue (filling queue size is 1): 18:18:11
Producer has enqueued a new box 4 to filling queue (filling queue size is 2): 18:18:12
Producer has enqueued a new box 5 to filling queue (filling queue size is 3): 18:18:13
Producer has enqueued a new box 6 to filling queue (filling queue size is 4): 18:18:14
Filler 2 finished filling the box 1: 18:18:15
Filler 2 put box 1 into packaging queue (packaging queue size is 1): 18:18:15
Packager 1 started packaging the box 1 (packaging queue size is 0): 18:18:15
Filler 2 started filling the box 3 (filling queue size is 3): 18:18:15
```

```
Producer has enqueued a new box 7 to filling queue (filling queue size is 4): 18:18:15
Filler 1 finished filling the box 2: 18:18:17
Filler 1 put box 2 into packaging queue (packaging queue size is 1): 18:18:17
Packager 2 started packaging the box 2 (packaging queue size is 0): 18:18:17
Filler 1 started filling the box 4 (filling queue size is 3): 18:18:17
Filler 2 finished filling the box 3: 18:18:22
Filler 2 put box 3 into packaging queue (packaging queue size is 1): 18:18:22
Filler 2 started filling the box 5 (filling queue size is 2): 18:18:22
Filler 1 finished filling the box 4: 18:18:23
Filler 1 put box 4 into packaging queue (packaging queue size is 2): 18:18:23
Filler 1 started filling the box 6 (filling queue size is 1): 18:18:23
Packager 2 finished packaging the box 2: 18:18:28
Packager 1 finished packaging the box 1: 18:18:28
Packager 2 started packaging the box 3 (packaging queue size is 1): 18:18:28
Packager 1 started packaging the box 4 (packaging queue size is 0): 18:18:28
Filler 2 finished filling the box 5: 18:18:28
Filler 2 put box 5 into packaging queue (packaging queue size is 1): 18:18:28
Filler 2 started filling the box 7 (filling queue size is 0): 18:18:28
Filler 1 finished filling the box 6: 18:18:30
Filler 1 put box 6 into packaging queue (packaging queue size is 2): 18:18:30
Filler 2 finished filling the box 7: 18:18:32
Filler 2 put box 7 into packaging queue (packaging queue size is 3): 18:18:32
Packager 2 finished packaging the box 3: 18:18:39
Packager 2 started packaging the box 5 (packaging queue size is 2): 18:18:39
Packager 1 finished packaging the box 4: 18:18:40
Packager 1 started packaging the box 6 (packaging queue size is 1): 18:18:40
Packager 2 finished packaging the box 5: 18:18:49
Packager 2 started packaging the box 7 (packaging queue size is 0): 18:18:49
Packager 1 finished packaging the box 6: 18:18:52
Packager 2 finished packaging the box 7: 18:19:02
End of the simulation ends: 18:19:02
```

Sample Run 6:

```
Please enter the total number of items:6
Please enter the min-max waiting time range of producer:
Min: 5
Max: 5
Please enter the min-max waiting time range of filler workers:
Min: 3
Max: 3
Please enter the min-max waiting time range of packager workers:
Min: 1
Max: 1
Simulation starts 19:10:48
Producer has enqueued a new box 1 to filling queue (filling queue size is 1): 19:10:53
Filler 1 started filling the box 1 (filling queue size is 0): 19:10:53
Filler 1 finished filling the box 1: 19:10:56
Filler 1 put box 1 into packaging queue (packaging queue size is 1): 19:10:56
Packager 2 started packaging the box 1 (packaging queue size is 0): 19:10:56
Packager 2 finished packaging the box 1: 19:10:57
Producer has enqueued a new box 2 to filling queue (filling queue size is 1): 19:10:58
Filler 1 started filling the box 2 (filling queue size is 0): 19:10:58
Filler 1 finished filling the box 2: 19:11:01
Packager 2 started packaging the box 2 (packaging queue size is 0): 19:11:01
Filler 1 put box 2 into packaging queue (packaging queue size is 1): 19:11:01
Packager 2 finished packaging the box 2: 19:11:02
Producer has enqueued a new box 3 to filling queue (filling queue size is 1): 19:11:03
Filler 1 started filling the box 3 (filling queue size is 0): 19:11:03
Filler 1 finished filling the box 3: 19:11:06
Filler 1 put box 3 into packaging queue (packaging queue size is 1): 19:11:06
Packager 2 started packaging the box 3 (packaging queue size is 0): 19:11:06
Packager 2 finished packaging the box 3: 19:11:07
Producer has enqueued a new box 4 to filling queue (filling queue size is 1): 19:11:08
Filler 2 started filling the box 4 (filling queue size is 0): 19:11:08
Filler 2 finished filling the box 4: 19:11:11
Filler 2 put box 4 into packaging queue (packaging queue size is 1): 19:11:11
Packager 2 started packaging the box 4 (packaging queue size is 0): 19:11:11
Packager 2 finished packaging the box 4: 19:11:12
Producer has enqueued a new box 5 to filling queue (filling queue size is 1): 19:11:13
Filler 1 started filling the box 5 (filling queue size is 0): 19:11:13
```

As you can see there is an inconsistency in the display order of the events occurred at the same time, as mentioned before they are acceptable.

```
Filler 1 finished filling the box 5: 19:11:16
Filler 1 put box 5 into packaging queue (packaging queue size is 1): 19:11:16
Packager 2 started packaging the box 5 (packaging queue size is 0): 19:11:16
Packager 2 finished packaging the box 5: 19:11:17
Producer has enqueued a new box 6 to filling queue (filling queue size is 1): 19:11:18
Filler 1 started filling the box 6 (filling queue size is 0): 19:11:18
Filler 1 finished filling the box 6: 19:11:21
Filler 1 put box 6 into packaging queue (packaging queue size is 1): 19:11:21
Packager 2 started packaging the box 6 (packaging queue size is 0): 19:11:21
Packager 2 finished packaging the box 6: 19:11:22
End of the simulation ends: 19:11:22
```

Sample Run 7:

As you can see there is an inconsistency in the display order of the events occurred at the same time, as mentioned before they are acceptable.

```
Please enter the total number of items:4
Please enter the min-max waiting time range of producer:
Min: 1
Max: 1
Please enter the min-max waiting time range of filler workers:
Min: 1
Max: 1
Please enter the min-max waiting time range of packager workers:
Min: 1
Max: 1
Simulation starts 23:26:51
Producer has enqueued a new box 1 to filling queue (filling queue size is 1): 23:26:52
Filler 2 started filling the box 1 (filling queue size is 0): 23:26:52
Filler 1 started filling the box 2 (filling queue size is 0): 23:26:53
Producer has enqueued a new box 2 to filling queue (filling queue size is 1): 23:26:53
Filler 2 finished filling the box 1: 23:26:53
Filler 2 put box 1 into packaging queue (packaging queue size is 1): 23:26:53
Packager 1 started packaging the box 1 (packaging queue size is 0): 23:26:53
Filler 1 finished filling the box 2: 23:26:54
Filler 1 put box 2 into packaging queue (packaging queue size is 1): 23:26:54
Packager 2 started packaging the box 2 (packaging queue size is 0): 23:26:54
Producer has enqueued a new box 3 to filling queue (filling queue size is 1): 23:26:54
Filler 2 started filling the box 3 (filling queue size is 0): 23:26:54
Packager 1 finished packaging the box 1: 23:26:54
Packager 2 finished packaging the box 2: 23:26:55
Producer has enqueued a new box 4 to filling queue (filling queue size is 1): 23:26:55
Filler 1 started filling the box 4 (filling queue size is 0): 23:26:55
Filler 2 finished filling the box 3: 23:26:55
Filler 2 put box 3 into packaging queue (packaging queue size is 1): 23:26:55
Packager 2 started packaging the box 3 (packaging queue size is 0): 23:26:55
Filler 1 finished filling the box 4: 23:26:56
Filler 1 put box 4 into packaging queue (packaging queue size is 1): 23:26:56
Packager 1 started packaging the box 4 (packaging queue size is 0): 23:26:56
Packager 2 finished packaging the box 3: 23:26:56
Packager 1 finished packaging the box 4: 23:26:57
End of the simulation ends: 23:26:57
```

Please see the previous homework specifications for the other important rules and the submission guidelines

Last, but not the least, you are highly recommended to use Windows for this homework. In multithreaded applications, there are several dependencies to the underlying operating system. Thus the behavior that you see while running in a Windows computer could be totally different than MacOS. Your code will be tested in Windows and if it does not work, even if it works using Mac, we may not re-evaluate your code due to hectic grading period at the end of the semester. To sum up, use Mac at your own risk.

Good Luck!
Albert Levi, Vedat Peran