

Northern Michigan University (Marquette Co, MI)

CS 222-01-14W: Data Structures

Program 3

Due: Wednesday 26 February 2014 10:00 A.M.

Print Queue

You are to write a program to emulate a print queue. The program will read commands from a file named "pg3cmds.txt" along the lines of

1:14 PRINT 5

This means that at 1:14 a print job is being sent to the queue that will take 5 minutes to complete. You may assume that the file will contain the commands in the correct format.

[h]h:mm<space>PRINT<space>[d]d . You may also assume that 1:00 is the earliest time and that 12:59 is the latest; the user will not cross over the next instance of 1:00. No print job will take more than 60 minutes to print, and its printing will not cross over the next instance of 1:00, either.

The user's input will end with the end-of-file.

The program's output will output the events of receiving the commands and completing the requests in chronological order. For example, if the input file were:

1:14 PRINT 5

1:16 PRINT 7

1:21 PRINT 2

The output would be

1:14 received 5-minute print request.

1:16 received 7-minute print request.

1:19 completed print request submitted at 1:14.

1:21 received 2-minute print request.

1:26 completed print request submitted at 1:16.

1:28 completed print request submitted at 1:21.

Notice that a print job can't begin until the previous one has completed. So the 5-minute job finished at 1:19, the 7-minute job finished at 1:26, and the 2-minute job finished at 1:28.

You must use a linked-list based queue to manage the print queue. It can be a single- or double-linked list, your choice, but you must use a head and a tail pointer for easy access to the list.

You must create a Queue class that contains your linked list. The Queue's public methods should **ONLY** contain a constructor, a destructor, and the ENQUEUE, DEQUEUE, and ISEMPY methods. Your main method can use the Queue class of course, but it cannot directly use the LinkedList or LinkedListNode classes, etc. How the Queue works should be invisible to the main.

Create a folder called PG3 in your CS222-01-14W folder. Place your .cpp and .h files (and only these files) into your PG3 folder. Place a (possibly empty) file or folder called "DONE" in PG3 when you are ready for your program to be graded.