

# Northern Michigan University (Marquette Co, MI)

## CS 222-01-14W: Data Structures

### Program 4

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

### Advanced Print Queue

Make a directory called “PG4” in the root of your CS222-01-14W folder. Place all the source code and include files in this directory. Place a (possibly empty) file called “DONE” into this folder when you are ready for your program to be graded.

Expanding on PG3, you are to program a more advanced print queue that allows the user to specify an integer priority for the job. Jobs with lower priority numbers print before jobs with higher priority numbers. If there is more than one entry with the same lowest priority number, the entry that was submitted first prints first.

You also have the ability to remove jobs from the queue before they are printed.

Your program should read all commands from the keyboard and print all responses to the screen.

The commands are:

SUBMIT <filename> <priority>

E.g. SUBMIT file1.pdf 100 will add file1.pdf to the print queue with priority 100. You may assume that the filename characters will be reasonable and will not contain a space. You may assume that the priority is a valid non-negative integer.

PRINT

This will print the next item from the print queue. Your output should specify what filename was printed.

REMOVE <filename>

Removes the filename from the print queue. If the filename appears more than once in the print queue, your program should remove all entries containing that filename.

EXIT

The program terminates.

Your program should give a meaningful response to EVERY valid command. E.g.:

```
SUBMIT file1.pdf 100
file1.pdf has been submitted to the print queue with priority 100
SUBMIT myblog.txt 50
myblog.txt has been submitted to the print queue with priority 50
PRINT
myblog.txt is printing
REMOVE myblog.txt
0 copies of myblog.txt has been removed from the print queue
REMOVE file1.pdf
1 copies of file1.pdf has been removed from the print queue
EXIT
Thank you for using the print queue
```

Your responses do not have to match the above exactly, but they must be meaningful.

Like all projects in this class, once the proper data structure is set up, the rest of the program should be reasonably straightforward.

You are to use a binary tree (as defined in class) implementing the Database interface, as specified with the various files.