

## CSCI1321 Final Review

This final will be formatted in a manner very similar to the midterm. The one exception is that it will only have 7 questions and 30% of your grade will come from a presentation on your semester project. What you need to do for that is discussed below after the list of topics. During the exam you will be able to have the Java API open on your computer, but no other references will be allowed.

### Topics:

**GUIs** – You need to understand how GUIs are constructed in Java and how events are handled. If I give you a piece of code you should be able to tell me what it does. I could also ask you to write a short piece of code that performs a certain task.

**Linked list based stacks, queues and priority queues** – You learned about stacks and queues in the first half of the semester using arrays. You need to know how they are implemented with linked lists as well and how the two implementations compare. Be able to write or trace such code. Also, you need to understand how a sorted linked list can be used to implement a priority queue and be able to write or trace it.

**Exceptions** – You must understand the concept of exceptions and how they work in Java (this includes both syntax and semantics). You should also be able to compare and contrast exceptions to other forms of error reporting. Know something about the different types of exceptions.

**Graphics** – You need to know how to do custom graphics in Java. What is the procedure? Also you should be able to write code to draw something extremely basic.

**Recursion** – Know how to write and trace recursive functions. You should also understand when recursive functions can/should be used to aid in problem solving. How do they compare to loops and when are they superior to loops?

**Binary Trees** – Know what a binary tree is and the things we can do with them. This includes sorted binary trees and the different traversals. You also need to understand what they are good for, their O speed, and when that breaks down. You need to be able to write code to do traversals or how to write basic functions to do things like find the height of a tree or count the number of leaves.

**Heaps** – You need to understand how a heap is implemented with a tree that has “heap ordering”. You should also know how to emulate a complete tree through the use of an array. Be able to write or trace code for this.

**Threads** – You need to understand the basic concept of what a thread is and how we create a thread in Java.

**Files and Streams** – Know how the stream library is set up in java.io and how we can use it for things like file access, binary data, and object serialization.

**Networking** – You need to have a basic understanding of how we do communication between computers in Java. You should also understand how this fits in with streams.

### Presentation Requirements:

Your presentation needs to cover three main things. You will do all of your presentation on the instructor’s machine in Linux. Each of these three pieces will count for roughly 10% of the exam grade.

1. Bring up your web documentation and describe your design. Tell us what your game is like and how each different class functions in it.
2. You should do a demo of your game on the instructor’s machine.
3. Describe the things that make your game unique. Talk about the features that were design challenges and how you got around them or significant bugs that you ran into. If you were to redo the project, what would you do differently? (In professional software development this is a significant question because systems can effectively be developed several times for the final version comes out.)