CSCI 470 Final Exam Study Guide

Swing Applications

- Know how to create a Swing-based Java application by extending the `JFrame` class.
- Know how to set the size and default close operation for an application, and how to make it visible.

Swing Components

Swing Classes and Methods you should study

- `JButton`
    - `JButton(String text)`
    - `void addActionListener(ActionListener l)`
- `JFrame`
    - `JFrame(String title)`
- `JToolBar`
    - `JToolBar()`
    - `JButton add(Action a)`
    - `void addSeparator()`

Events and Event Handling

Event Classes you should study

- `ActionEvent`
    - `String getActionCommand()`
    - `Object getSource()`

Interfaces

- `ActionListener`
    - `void actionPerformed(ActionEvent e)`

Multithreading

- Be able to describe the two techniques for creating a new thread of execution:
    - Extending the `Thread` class
    - Implementing the `Runnable` interface
- Know the six possible states of a thread's life cycle (listed in `Thread.State`) and be able to describe the circumstances under which a thread transitions from one state to another.
    - `NEW`
    - `RUNNABLE`
    - `BLOCKED`
    - `WAITING`
    - `TIMED_WAITING`
    - `TERMINATED`
- Be able to explain the terms *thread interference* and *memory consistency error* and why they can be problems when writing multithreaded code.
- Know how to *synchronize* a method or block of code to prevent thread interference and memory consistency errors.

- Be able to explain what an *intrinsic lock* (or *monitor lock*) is and how it is used by synchronized code.
- Be able to explain what the term *deadlock* means and why it might happen.
- Be familiar with the preferred technique for stopping a background thread in response to a user action such as clicking a button.

Thread Methods you should study

Know how the following methods are used and the effect they have when called.

- Methods of the `Thread` class:
    - `void setPriority(int newPriority)`
    - `static void sleep(long milliseconds)`
    - `void start()`
    - `static void yield()`
- Methods of the `Object` class:
    - `void notify()`
    - `void notifyAll()`
    - `void wait()`
    - `void wait(long timeout)`
- Methods of the `Runnable` interface:
    - `void run()`

Synchronization

- Know what a *monitor lock* is and be able to describe how a thread may use an object's monitor lock to enforce mutual exclusion.
- Be able to use the `synchronize` keyword to synchronize a method or a set of statements.
- Be able to describe what type of operations and data require synchronization.
- Be able to define the term *deadlock* and given an example of how it might occur in a multithreaded Java program.

Painting

- Be able to explain the connection between calling `repaint()` for a component, the `JComponent` method `paint()` and the method `paintComponent()` that you override in an application.
- Know how to use the following drawing methods of the `Graphics` class:
    - `boolean drawImage(Image img, int x, int y, ImageObserver observer)`
    - `void drawLine(int x1, int y1, int x2, int y2)`
    - `boolean drawString(String str, int x, int y)`
- Know how to obtain the dimensions of a Java component using the method `getSize()`.

Fonts

- Understand the difference between a *logical font* and a *physical font.*

Networking

- Know how to create a client `Socket` to attempt to connect to a server.
- Know how to create a `ServerSocket` that will listen for and accept connections from a client.
- Know how to obtain the input and output streams for a `Socket`.

Object Serialization

- Know how to implement the `Serializable` interface for a class.
- Know how to create an `ObjectInputStream` and an `ObjectOutputStream` from a `Socket`'s input and output streams. What errors can occur during this process?
- Know how to read an object or write an object using those streams.

JDBC

- You do **not** have to study the code to load the database `DriverManager` class. You should understand that the `DriverManager` is used to obtain a `Connection` object. However, any coding questions will assume you already have a valid `Connection` object.
- Know how to create a `Statement` object from a `Connection`.
- Know how to create a `PreparedStatement` object from a `Connection`.
- Know how to set integer and string parameters for a `PreparedStatement`.
- Know how to execute an update to add or delete a table row using a `Statement` or `PreparedStatement`.
- Know how to execute a basic query using a `Statement` or `PreparedStatement` and how to process the `ResultSet` that is returned. For example, you should be able to write the code to execute and process a query like “retrieve and print the name field of every row in the database”.