CS3733 Midterm Examination

Answers:

- 1. Incremental Model
 - (a) S1: If a project can be broken up into a layered set of requirements, the innermost being the core set of requirements, then the project can be released in stages, thus getting the product to market faster.
 - W1: Let's say we've released the first increment, and are halfway through the second. If a problem occurs (say in the core functionality) that forces a complete rework of the system, the entire model is thrown off. It doesn't react to failure gracefully.
 - (b) Rapid Prototyping
 - S1: RP helps the client get a better sense of the final system by evaluating and refining the initial requirements. There is at least one moment before the final product release where the client can interact with the developers.
 - W1: The prototype is not a formal requirements document, so it won't cover the entire specification, and more work needs to be done on specifying other aspects of the system. The RP is only good for the User Interface and expected functionality.
 - (c) The Spiral model has more interaction between the client and developer because at every iteration through the cycle, the client can assess risks and evaluate whether to go ahead with the project.
- 2. An **Interface** specifies a set of **Methods** for a class. A class can **Implement** multiple **Interfaces**, but each **Method** belongs to exactly one **Interface**.
- 3. Unnecessary Coupling
 - I1: It reduces the chance that an object will be reused, since unnecessary coupling increases dependencies with objects in other classes.
 - I2: It increases maintenance costs by making it easier for changes to affect multiple objects.
- 4. CRC question

The CRC diagram determines which classes collaborate with other classes (and on which method). If several classes form a close collaboration with each other, and with no other classes, then these classes form a logical unit, which may be a potential subsystem.

The transitive closure of a CRC diagram may identify candidate subsystems, much like object-oriented analysis methods suggest starting with nouns as candidate classes.

5. Stamp coupling

Yes. If obj2 has any public attributes defined in its interface, then object1 can access these directly. This is another reason why attributes should be private to each class. Alternatively, if the class for object1 is declared as a *friend* (C++ term) to the class for obj2, it can access the underlying implementation.

- 6. Match concepts with their definition:
 - (1) The ability to reuse existing modules
 - (2) The feature that changes are localized to individual modules
 - (3) Suppressing implementation details of a module from other modules.
 - (4) The ability to handle error conditions within a module without revealing the details to other modules.
 - (3) (a) Information Hiding (1) (b) Modular composition
 - (4) (c) Modular protection (2) (d) Modular continuity
- 7. PartSortList behavioral model

8. Class Hierarchy for Mathematical concepts

Note: These answers are not the only answers for this exam, but are indicative of the level of detail I am looking for.