

## Notes and recommendations about the ICS 52 Homework 2 Class Diagram:

- Classes are based on the Official SubDo Requirements.
- The diagram is a (simplified) version of the UML class diagram. (See discussion overheads.)
- Keep class boxes reasonably sized by generally listing no more than three fields and no more than five methods. Select the fields and methods which convey the most information about what the class is and does. Of course, in your Class Interface Specification you'll list **all** public methods in the class's interface.
- Indicating Aggregation and Composition relationships between classes is optional. It's fine to use only Association relationships.
- Use the Generalization relationship when (and if) appropriate. Make sure the arrowhead is pointing in the right way – from the specialized class to the more general class.
- Most Association relationship lines should be labeled with a word or two describing the relationship. Use an arrowhead if that makes the label more easily understood.
- Indicate cardinality for most relationships.
- Use of visibility / access modifiers (e.g. - for private, + for public) is optional.
- Indicating return values and parameters of methods is optional.
- Think carefully about whether a class name should be singular or plural.
- As a rule, don't include boxes for Java library classes (e.g. ArrayList).
- As a rule, don't include boxes for the special kind of class called an "interface" in Java; and don't include boxes for Java inner classes.
- As a rule, use one or two class boxes in the Class Diagram to represent the user interface. If you have a "GUI" class box, it should have fields and methods, just like the other class boxes.
- Your design should encompass all implementation phases (not just the first phase), including the data conversion described in 3.8.
- Don't try to make this diagram say more than it is designed to say. Don't indicate flow of control, passing of data, or control hierarchy.