**OOP2 Hand-in Project Assignment, Semester 1, 2017**.

First CA Practical 20%-30**% project**

**Due date**:

Project must be demonstrated to the lecturer week beginning 27st November. You should have all of your documentation available at your presentation.

**Project Specification** (minimum requirements unless agreed with Lecturer):

* At least three classes required (at least 2 of which are instantiable)
* At least one of the classes is a GUI class, having a menu system plus at least three other ‘J’ components
* Each instantiable class should model a different aspect of a system (for

example a Book and a Student, as part of a Library system).

* The application should contain a ‘substantial’ processing element (an algorithm for doing something!). At a minimum, your system should allow the user to process at least one transaction involving an object from each class (e.g. process the return of a Book by a Student).
* Demonstrate superclass/subclass inheritance, an Interface and/or composition/aggregation
* System should allow you to add, display and hold in memory in an appropriate data structure (yet to be covered) instances of each instantiable class,
* Save the data structure to file (yet to be covered) and load it up again.
* At least one class should be fully commented with Javadoc comments
* The evolution of the project must be clearly demonstrated. Using git/GitHub as a version control system is mandatory.

**Note:**

Classes used as examples in class or featuring on lab sheets are not acceptable but can be used to form the basis of your classes. Each student should choose a different topic unless otherwise agreed with the lecturer. All choices subject to lecturer approval.

**Documentation to be submitted:**

Requirement Specification (word), more than one paragraph, less than one page

Class diagrams for all classes (word), done up in Visio or appropriate tool

Outline VOPC diagram for the GUI application (word)

Program code (.java)

Javadoc output for the Javadoc commented class

GitHub logs

\*\*Declaration of Originality

You will also be asked to give a quick demonstration of your working project on the week beginning 27th November, **No demo = no marks. As part of your demonstration you will be asked about your code.**

**Indicative marking scheme: Projects will be graded using the following criteria**:

* Scope/Complexity- safe, narrow, broad. (40%)
* Code features (classes, methods, structure choices, inheritance/composition, going beyond what we have covered) (20%)
* Quality of code (e.g. naming conventions, comments, indentation, methods, usability, efficiency) (10%)
* Presentation (functioning, usability, design) (20 %)
* Quality of Submitted Documentation (10%)