

# CS1410 Java Program Development

Level: 1

Credits: 20

Teaching Period: 2

Module Tutor: Prof IT Nabney

## Aims

This module will build on the foundations laid in CS1310. It will introduce inheritance and software structuring concepts that provide the object-oriented approach to software development with much of its power. Students' programming capability will be enhanced through substantial practical work and increased knowledge of software development methodology.

## Content

### Object-oriented Structure

- Inheritance: subtyping, inclusion polymorphism, polymorphic collections
- Method overriding and dynamic method dispatch
- Abstract methods and classes, interfaces
- Visibility and scope: inner classes and packages

### Object-oriented Design

- Client-server interaction: Abstract Data Types; error handling and exceptions, application to text-based and file I/O
- Use of UML to document design: class and sequence diagrams
- Introduction to design patterns

### Java Frameworks

- GUI programming: Swing components, layout, event-driven execution, event listeners
- Collections framework: use of JCF containers and algorithms
- I/O: text-based, file, serialisation

### Software Development

- Software lifecycle: analysis, design, coding, testing and debugging, documentation. Iteration and refactoring
- Transition from an introductory to a standard development environment
- Software engineering for groups
- Extensive case studies

## Teaching

Lectures: 33 hours (3 hours/week)

Tutorials: 9 hours (1 hour/week)

Practical classes: 20 hour (One 2-hour session each week)

## Assessment

Written exam: 70% (2 hours, May/June)

Practical assignment: 20% (group-based)

Lab work: 10% – This assessment element will require attendance at scheduled classes

## Module outcomes

| <i>What the student should gain from successful completion of the module</i>  | <i>Teaching/Learning Methods</i>  | <i>Assessment Methods</i>      |
|---|---|--------------------------------|
| <i>Knowledge and Understanding</i>  |   |                                |
| Define class structuring concepts used in object-oriented software.<br>Describe Java language support for good system design.<br>Define the purpose and (at an introductory level) the structure of a systematic software process   | Lectures and recommended reading  | Exam                           |
| <i>Intellectual Skills</i>  |   |                                |
| Analyse requirements and design appropriate software solutions utilising object-oriented client-service modularity.<br>Follow a systematic software development process and evaluate outcomes.  | Lectures, with practical demonstrations, and practical work in lab classes and individually | Exam and coursework            |
| <i>Professional/Subject-Specific Skills</i>   |   |                                |
| Use standard Java class libraries including GUIs, containers and I/O.<br>Undertake systematic testing making use of appropriate tools.<br>Use UML to document static and dynamic aspects of a system design.<br>Implement medium-sized software systems in Java using a standard IDE. | As above, plus use of online Java resources   | Exam, lab work, and coursework |
| <i>Transferable Skills</i>  |   |                                |
| Evaluate and report on system development.<br>Work cooperatively in a team.   | Practical exercises   | Coursework                     |

## Learning resources

Barnes & Kolling: *Objects First with Java – A Practical Introduction Using BlueJ (fourth edition)*, Pearson Education, 2008

## Other study requirements to take this module

CS1310 Java Programming Foundations