CM1210

Object Oriented Java Programming

LECTURE 1

MODULE INTRODUCTION

Module Description

The aim of this module is to provide students with the fundamental programming and problem solving skills necessary to develop an understanding of **Object Oriented Programming**. It will also introduce fundamental **data structures and algorithms** used in Computational systems. This knowledge will serve as a foundation for future taught modules. Focus on **Object Oriented Design** will be supported through the use of core **worked examples** that will be built in some of the **lab-based sessions**.

Learning Outcomes

Weeks 1 to 5 (inclusive) [Java Language] [Matt Morgan]

1. Develop an Object-Oriented program that has input and output functionality and that is event driven.
2. Show fluency in selecting and using basic components in the Java language.
3. Show an understanding of the theoretical underpinnings of the Java language.

Weeks 6 to 11 (inclusive) [Data Structures & Algorithms] [Yuhua Li]

4. Implement basic data structures and algorithms.
5. Analyse and describe the performance of data-structures and algorithms.
6. Deliver technical reports on Object-Oriented systems.

Weeks 1 to 5 Delivery

Module Delivery – Weeks 1 to 5 (inclusive) [Java Language]

Lectures [Matt Morgan]

- There will be **ONLY ONE** Lecture for the Java Language part of this module (i.e. in weeks 1 to 5 inclusive) and this is it!!
- **ALL** material will be delivered through **Laboratory Sessions**.
- These sessions will be supported by **Exercise Classes** (lab based) and **further material**.

Laboratory Sessions (Weeks 1 – 5)* [Matt Morgan]

Wednesday 09:00 to 10:50 [GROUP A]
Wednesday 11:10 to 13:00 [GROUP B]

Exercise Classes (Weeks 2 - 6)* [PhD/Undergraduate Students]

Tuesday 15:10 to 16:00 [GROUP A]
Tuesday 16:10 to 17:00 [GROUP B]

Further Material

- Delivered through a range of mechanisms (e.g. PDF, Code and Podcasts).

* Each student should attend ONE Laboratory Session AND ONE Exercise Class each week.

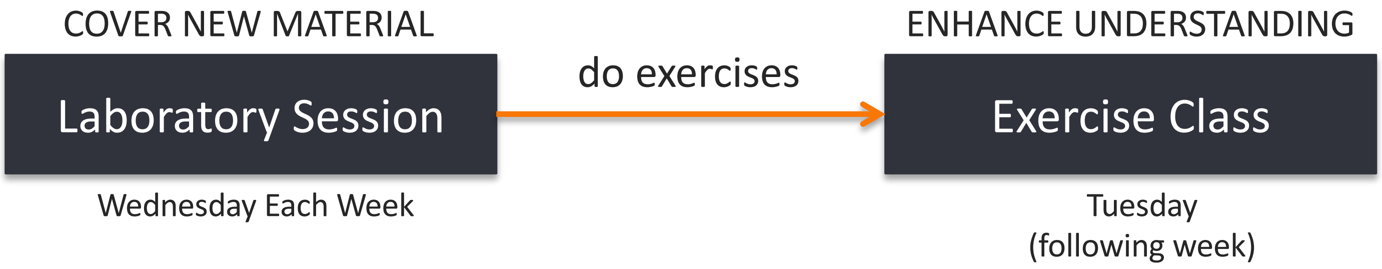
Week 1 to 5 Materials

Learning Materials

- With a view to saving a small forest, **ALL** module materials will be **electronic**.
- If you prefer to work from printed material, remember that you can use the printers in any of our computer laboratories.

Learning Central

- Learning Central will be the **central repository** for all learning materials.
- Material from the 2 hour Laboratory Sessions, on Wednesday mornings, will become available each week by **13:00 Thursday** (i.e. once both sessions have taken place).
- Material for the following weeks Exercise Classes on Tuesdays, will also be provided by **11am every Wednesday** morning. The exercise sheets will cover the material covered in the "Taught" Laboratory Sessions. Undertaking the exercises, will enhance your understanding. **Solutions** to exercises **WILL BE PROVIDED**.



COVER NEW MATERIAL

Laboratory Session

Wednesday Each Week

do exercises

ENHANCE UNDERSTANDING

Exercise Class

Tuesday
(following week)

Help Session Week 6

Laboratory Sessions (Weeks 6)

[Matt Morgan]

Wednesday 09:00 to 10:50 [GROUP A]
Wednesday 11:10 to 13:00 [GROUP B]

This will be used as a help session for the FIRST of TWO courseworks that you will undertake in the **CM1210** module.

Weeks 7 to 11 Delivery

Module Delivery – Weeks 7 to 11 (inclusive) [Data Structures & Algorithms]

Lectures (Weeks 7 - 11) [Yuhua Li]

- TWO lectures each week:

Tuesdays @ 13:10 in N/4.07
Thursdays @16:10 in S/1.32

Exercise Classes (Weeks 7 - 11) [PhD/Undergraduate Students]

Wednesday 15:10 to 16:00 [GROUP A]
Wednesday 16:10 to 17:00 [GROUP B]

Syllabus (1)

Java Fundamentals

- Introduction to data types, identifiers, variable declarations, constants, comments, program output and simple arithmetic operations.
- Programming style conventions.

Decision Structures

- Relational operators/expressions
- Control the flow of a program with if /switch statements and conditional/logical operators
- Comparing Objects and primitive types

Control Structures & Files

- Control structures
- Counters, accumulators, running totals, sentinels and other application/related topics are explored
- Basic read/write file operations

Methods

- Explore the various ways of writing methods
- Functional decomposition

Syllabus (2)

Classes

- Class fields/methods, constructors, overloading, packages and imports.
- Static method/fields, interaction between objects, passing objects as arguments and returning objects.
- Aggregation and enumerated types
- Garbage collection

Input Processing

- Introduction to wrapper Classes
- Character Class, String Methods, tokenizing Strings
- Wrapper Classes for numeric data

Inheritance

- Chains of inheritance
- The Object class
- Polymorphism
- Abstract Classes/Methods

Exceptions Handling

- Handling and Throwing Exceptions

Syllabus (3)

Implementing basic data structures

- Linked Lists
- Stack
- Queue

Implementing simple algorithms

- Simple search algorithms
- Understanding the performance of data structures and algorithms
- Creating professional reports communicating technical details of object oriented software and the data structures & algorithms underpinning it
- Using IDEs for software development

Assessment

Assessment Breakdown

Type	%	Title	When
Practical-Based Assessment	50	Java Implementation Skills	Week 5 to Week 9
Written Assessment	50	Problem Solving Exercise	Week 8 to Week 11

Expectations

What Should You Be Doing

- A **10 Credit** module assumes a total of **100 hours** expended (includes contact time, coursework, exams, etc...)
- The following table provides you a guide to the time expectations of this module. **NOTE:** The numbers given relate to an "average" student and should be adjusted accordingly.

Activity	Hours Per Week	Total Hours
Laboratory Sessions	2	22
Exercise/Feedback Classes	1	11
Coursework		46
Working In Own Time	11	121
		200

NOTE: CM1210 IS A 20 CREDIT MODULE – SO THESE FIGURES ARE DOUBLE ☺

Reading and Resources

Recommended Reading

Recommended: **Starting Out with Java**, Tony Gaddis, Pearson
6th Edition, 2016 – updated for Java 8 (1.8)
Recommended: **Big Java**, Cay S. Horstmann, Wiley
4th Edition, 2010 – updated for Java 7 (1.7) onwards
For a free perspective: **Think Java**

- 5th edition, Allen B. Downey , Green Tea Press, 2012
- 6th edition (Draft), Allen B. Downey and Chris Mayfield , Green Tea Press, 2016

http://www.greenteapress.com/thinkapjava/

The Java™ Tutorials

https://docs.oracle.com/javase/tutorial/

Java™ Platform, Standard Edition 8 API Specification

http://docs.oracle.com/javase/8/docs/api/

NOTE: ANY books covering Java 7 or 8 are appropriate for this module.

Developing in Java

To develop and compile applications for Java SE, you need the **Java Development Kit** (JDK), which you can download from:

http://www.oracle.com/technetwork/java/javase/downloads/index.html

This includes the Java compiler (**javac**), Core API packages, and the Java Virtual Machine (**java**)

It is also possible to obtain the Java Virtual Machine without the compiler:

http://www.java.com/en/download/index.jsp

This enables users to execute Java applications that have been compiled elsewhere (i.e., by developers)

E.g., Android phones have Java virtual machines to execute apps

NOTE: Download the Java SE 8 JDK – current version is **Java SE 8u201/8u202**

Text Editors & IDEs

Text Editors

- Initially, we will be using a text editor to write code and the command line to compile/execute applications.
- You can use ANY text editor to do this. Just use your favourite or the default on your Operating System.
- WHY A TEXT EDITOR? It will allow you a much deeper understanding of things that are abstracted away when using an Integrated Development Environment.

Integrated Development Environment

- Professional programmers use one of the Integrated Development Environments (IDEs), which include an editor, a compiler, type-ahead help, a debugger, and a lot more.
- There several popular Java IDEs, such as Eclipse, NetBeans, IntelliJ IDEA, and RAD. Some are free, and some are not.
- We will later move to using the Eclipse IDE.
- Eclipse IDE is an open-source product that was originally created with a substantial code donation by IBM to the Java community, and from that moment Eclipse was a 100% community-driven product.

Eclipse IDE for Java Developers can be downloaded from:
http://www.eclipse.org/downloads/packages/eclipse-ide-java-developers/mars1

Ongoing Feedback

I Want Your Feedback

Feedback is **EXTREMELY IMPORTANT TO ME**. It is fundamental to my understanding of what is going well and what is not!

Every cohort of students is **DIFFERENT**. What has worked for one group, may not for another.

This module is being taught in a radically different way.

Are there things you like?, Things you don't like?, Suggestions of any kind? I really would like to hear from you.

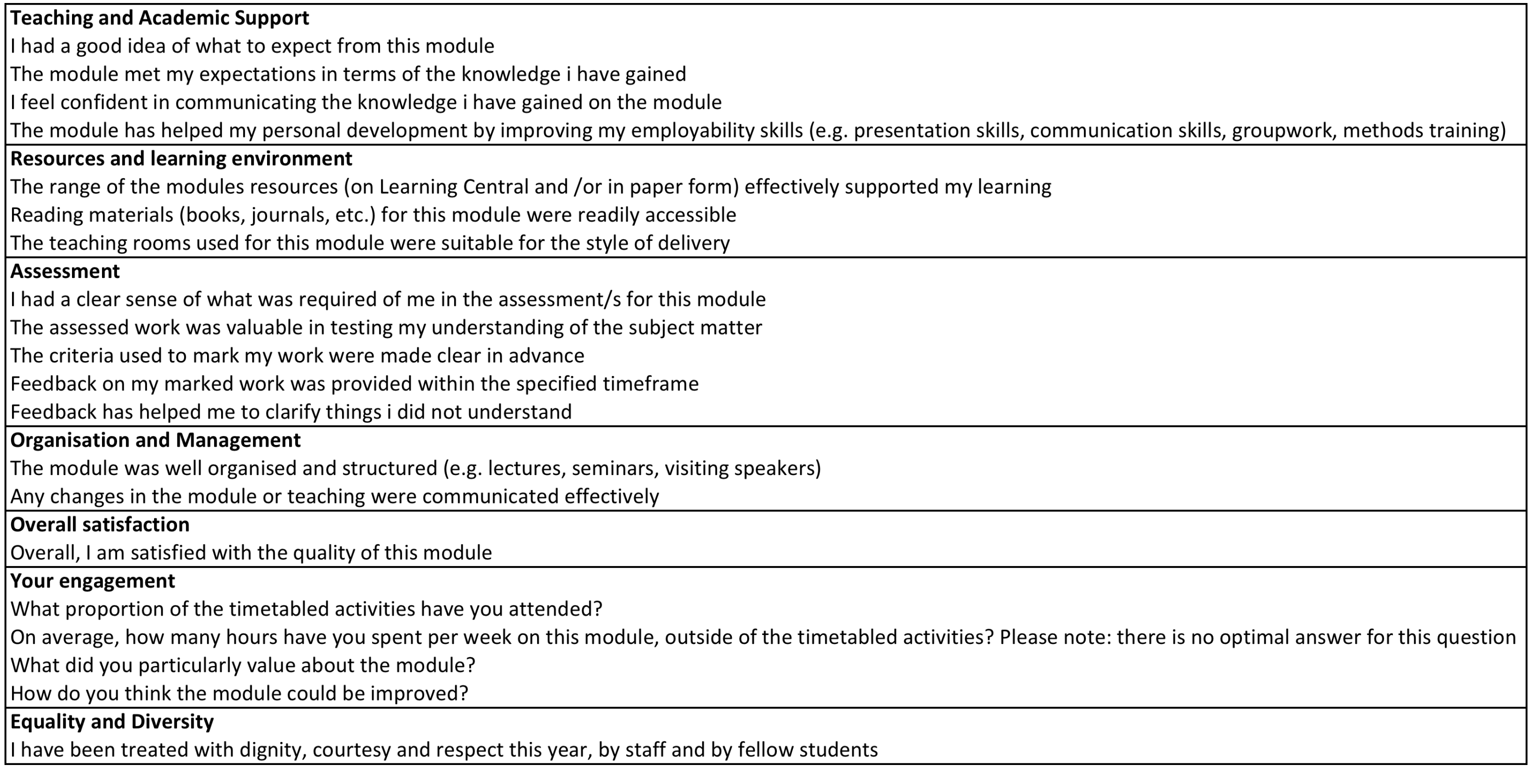
NOTE: I am of course happy for you to just come and have a chat face to face ☺

End OF Module Feedback

Feedback On Completion

At the end of the module, you will be asked to complete an online questionnaire for this module.

You will answer the series of questions below:



Teaching and Academic Support
I had a good idea of what to expect from this module
The module met my expectations in terms of the knowledge i have gained
I feel confident in communicating the knowledge i have gained on the module
The module has helped my personal development by improving my employability skills (e.g. presentation skills, communication skills, groupwork, methods training)
Resources and learning environment
The range of the modules resources (on Learning Central and /or in paper form) effectively supported my learning
Reading materials (books, journals, etc.) for this module were readily accessible
The teaching rooms used for this module were suitable for the style of delivery
Assessment
I had a clear sense of what was required of me in the assessment/s for this module
The assessed work was valuable in testing my understanding of the subject matter
The criteria used to mark my work were made clear in advance
Feedback on my marked work was provided within the specified timeframe
Feedback has helped me to clarify things i did not understand
Organisation and Management
The module was well organised and structured (e.g. lectures, seminars, visiting speakers)
Any changes in the module or teaching were communicated effectively
Overall satisfaction
Overall, I am satisfied with the quality of this module
Your engagement
What proportion of the timetabled activities have you attended?
On average, how many hours have you spent per week on this module, outside of the timetabled activities? Please note: there is no optimal answer for this question
What did you particularly value about the module?
How do you think the module could be improved?
Equality and Diversity
I have been treated with dignity, courtesy and respect this year, by staff and by fellow students

Please **READ** these. If you feel that I am not achieving any of these things, please get in touch.

Questions?