

Abbreviated Module Guide 2015/16

| CI4100 Programming 1 | | | | |
|-----------------------------|----------------|--------|-----------|--|
| Staff | Name | Room | Extension | Contact email and consultation hours |
| Module leader | Paul Neve (PN) | SB3034 | 67041 | paul@kingston.ac.uk Office hours may differ from week to week due to variations in timetabled classes. Consult Studyspace for the most up to date details. |

| | |
|--|--|
| Teaching schedule information | You will be able to access your timetable for the 2015/16 academic year via the University mobile app or via OSIS (the Online Student Information System). More information on all aspects of timetabling can be found on the MyTimetable pages on MyKingston. |
| <p>Each student is expected to attend at least</p> <p>1 x 2 hour lecture weekly 1 x 2 hour workshop weekly</p> <ul style="list-style-type: none"> • Times and locations of lectures workshops vary throughout the year by cohort and group. • Please consult OSIS for times and locations for your regular lectures and workshop sessions. • Note that it may take some time for the published online timetable on OSIS or the MyTimetable pages to catch up with any changes. Thus you should ALWAYS check Studyspace the start of each week. Changes will also be emailed to your university email account. If Studyspace and/or an email sent to you contradicts the timetabling app, Studyspace and/or email should be considered the authoritative source. | |

| In-course assessment These dates are indicative. Consult Studyspace for up-to-date information on assessment. | Type | % | Due dates | Feedback |
|---|---|-----|---|-------------------------------|
| | Weekly workshop activities | 60% | Unit 1: Fri 6 th November Unit 2: 11 th January Unit 3: 22 nd February Unit 4: 23 rd April | Immediate via NoobLab |
| | In-class "spot" tests plus one-shot questions during lectures | 40% | Spot Test 1: Week of 26 th October Spot Test 2: Week of 7 th December Spot Test 3: Week of 8 th February Spot Test 4: Week of 4 th April | Immediate, following the test |

MODULE SUMMARY

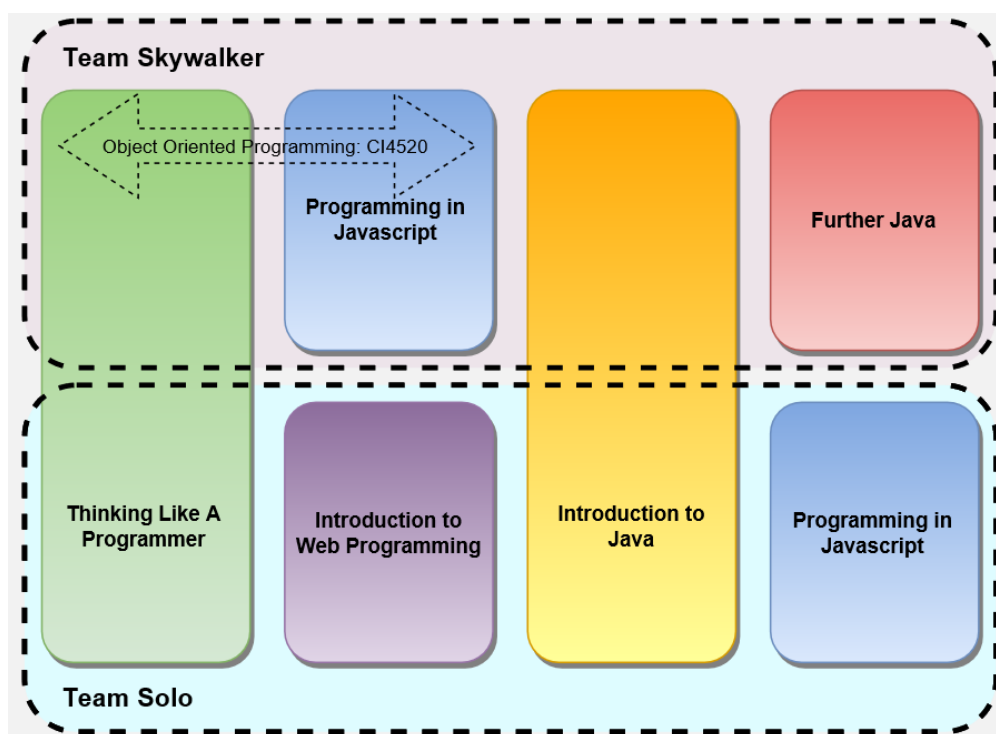
Welcome to the module!

The aim of the module is to provide a foundation for all programming activities that follow in subsequent years of your course. We do not assume that you have previous experience of programming; we start from the very beginning. We try to develop your ability to break problems down and “think like a programmer” before we break your brain with complex programming languages and more advanced concepts.

There is a misconception that programming is “hard” or “boring”. My job is to persuade you that it’s not. We try to make the material as engaging and fun as possible and we make use of our own homegrown NoobLab environment to do so.

Students doing Object Oriented Programming will do the units Thinking Like A Programmer and Programming in JavaScript with Paul before Christmas. They will then go off to do C++ with Ahmed.

Students doing Programming 1 will do four out of five possible units. At the end of Thinking Like A Programmer we will take a look at your progress. You will then be allocated into either Team Skywalker or Team Solo. This will then determine which units you will do as follows:



During Enrichment Activity Week (week 6 of term) I will look at your progress. Those of you who have demonstrated an aptitude for the material in Thinking Like A Programmer will join Team Skywalker and will focus more on Java with on Object Oriented principles in the latter part of the module. Those of you who need a little more support will not do as much Java, and you’ll look at Web application programming instead as a member of Team Solo. Both routes through the module carry the same assessment weight and will set you up for Programming 2 in the second year. Neither route is “better” and it is possible to give 100% for the module

regardless of which route you end up following. What we want to do is to give everyone the best possible chance of success, and the best possible chance to maximise their potential.

LECTURE PROGRAMME

Each unit consists of five lectures with associated workshops. This is an *indicative* schedule and may be subject to change.

Thinking Like a Programmer (Both groups)

| Week of | Subject |
|---------|------------------------------------|
| Sep 28 | Introduction to the Module |
| Oct 5 | Fundamental programming constructs |
| Oct 12 | Booleans and more on functions |
| Oct 19 | Moving to “real” code |
| Oct 26 | In Class Test |

Programming in Javascript (both groups – Team Skywalker do this as their second unit, Team Solo do this as their last unit)

| Week of (Skywalker) | Week of (Solo) | Subject |
|---------------------|----------------|--|
| Nov 9 | Feb 22 | Introduction to Javascript |
| Nov 16 | Mar 29 | Functions and variable |
| Nov 23 | Mar 7 | HTML and the Document Object Model (or “everything you’ve learned is a lie”) |
| Nov 30 | Mar 14 | Events on the DOM and creating interactivity |
| Dec 7 | Apr 4 | In Class Test |

The Basics of Web Programming (Team Solo)

| Week of | Subject |
|---------|---|
| Nov 9 | The basics of HTML |
| Nov 16 | Links, images, forms and multi-page sites |
| Nov 23 | Introduction to PHP |
| Nov 30 | Reading form data with PHP |
| Dec 7 | In Class Test |

Introduction to Java (both groups)

| Week No | Subject |
|---------|--|
| Jan 11 | The basics of the Java language |
| Jan 18 | Conditional and loop constructs / arrays |
| Jan 25 | Introduction to object orientation (or “everything you’ve learned is a lie”) |
| Feb 1 | “Madness in the Methods”: parameters, return values and constructors |
| Feb 8 | In Class Test |

Object Oriented Programming in Java (Team Skywalker only)

| Week No | Subject |
|---------|--|
| Feb 22 | Encapsulation and packaging |
| Mar 29 | Arrays of Objects / Inheritance |
| Mar 7 | Collections |
| Mar 14 | Java programming in the real world: IDEs and bringing it all together |
| Apr 4 | In Class Test |

WORKSHOP/SEMINAR/TUTORIAL PROGRAMME

The 2 hour lecture each week will be supplemented by a 2 hour hands-on workshop session. Please see Studyspace and OSIS for details of the location and times.

During enrichment weeks, we will endeavour to provide additional support available in the form of optional “codebash” sessions. Look for announcements on Studyspace.

READING LIST

The core texts for the module this year are:

Jon Duckett: HTML & CSS: Design and Build Web Sites

- http://www.amazon.co.uk/HTML-CSS-Design-Build-Sites/dp/1118008189/ref=la_B001IR3Q7I_1_1?s=books&ie=UTF8&qid=143920

Jon Duckett: JavaScript and JQuery: Interactive Front-end Web Development

- http://www.amazon.co.uk/JavaScript-JQuery-Interactive-Front-end-Development/dp/1118531647/ref=la_B001IR3Q7I_1_3?s=books&ie=UTF8&qid=1439202657&sr=1-3

Lynn Beighley and Michael Morrison: Head First PHP and MySQL

- <http://www.amazon.co.uk/Head-First-MySQL-Lynn-Beighley/dp/0596006306>

Cay Horstmann: Big Java (late objects)

- http://www.amazon.co.uk/Big-Java-Late-Objects-Horstmann/dp/1118087887/ref=sr_1_1?s=books&ie=UTF8&qid=1442237249&sr=1-1&keywords=big+java+late+objects

Do note that programming is a *practical*, hands-on activity that is fast-moving in terms of best practice. Do not make the mistake of thinking that there is a “holy grail” textbook out there that turns people into amazing programmers just by the act of possessing it! Textbooks can quickly become out of date. The best way to become a proficient programmer is by programming – not by reading a textbook! **I do NOT recommend spending any serious amounts of money on textbooks!**