# Kingston University London

Faculty of Science, Engineering and Computing

### Abbreviated Module Guide 2015/16

CI4100 Programming 1				
Staff	Name	Room	Phone	Contact email and consultation hours
Module leader	Paul Neve (PN)	SB3013	020 8417 7041 KU Ext: 67041	paul@kingston.ac.uk Consultation hours: Thursday 10am-noon Friday 10am-noon

In weeks 1-5 lectures take place at 2pm on Monday afternoons in PRSB2025 (the Roberts Lecture Theatre). Workshops take place in SB2022/3 on Wednesday mornings; you will be allocated into either the 9am or 11am slot.

After the first enrichment week, lecture times and locations will change and will depend on which group you are assigned to.

- Please consult OSIS, Studyspace and your KU email regularly to confirm times and locations of lectures and workshop sessions. These are subject to change!
- Note that it has been known for the published online timetable on OSIS or the MyTimetable pages to take time 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. Check this too!
- If Studyspace, an email, or something your lecturer tells you sent to you contradicts the timetabling app or OSIS, **Studyspace, email or the lecturer should be considered the authoritative source!**

In-course	Туре	%	Due dates	Feedback
assessment These dates are indicative. Consult	Weekly workshop activities	60%	Unit 1: Mon 7 <sup>th</sup> November 12:01am Unit 2: Mon 9 <sup>th</sup> January 12:01am Unit 3: Mon 20 <sup>th</sup> February 12:01am Unit 4: Fri 31 <sup>st</sup> March 5pm	Immediately via NoobLab
Studyspace for up-to-date information on assessment.	Clicker questions during lectures	40%	Every week	Immediately, discussed in class after the question

#### 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 get 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 26	Introduction to the Module
Oct 3	Fundamental programming constructs
Oct 10	Booleans and more on functions
Oct 17	Moving to "real" code
Oct 24	Orange Event

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

Week of	Week of	Subject
(Skywalker)	(Solo)	
Nov 7	Feb 22	Introduction to Javascript
Nov 14	Mar 29	Functions and variables
Nov 21	Mar 7	HTML and the Document Object Model (or "everything
		you've learned is a lie")
Nov 28	Mar 14	Events on the DOM and creating interactivity
Dec 5	Apr 4	Orange Event

#### The Basics of Web Programming (Team Solo)

Week of	Subject
Nov 7	The basics of HTML
Nov 14	Links, images, forms and multi-page sites
Nov 21	Introduction to PHP
Nov 28	Reading form data with PHP
Dec 5	Orange Event

#### Introduction to Java (both groups)

Week of	Subject
Jan 09	The basics of the Java language
Jan 16	Conditional and loop constructs / arrays
Jan 23	Introduction to object orientation (or "everything you've learned is a
	lie")
Jan 30	"Madness in the Methods": parameters, return values and
	constructors
Feb 06	Orange Event

#### **Object Oriented Programming in Java (Team Skywalker only)**

Week of	Subject
Feb 20	Encapsulation and packaging
Feb 27	Arrays of Objects / Inheritance
Mar 6	Collections
Mar 13	Java programming in the real world: IDEs and bringing it all together
Mar 20	Orange Event

#### 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

Please note the books given in the Module Descriptor are indicative and represent an old version of the module. The core texts for the module this year are below.

However, be warned: I give a reading list solely because I am required to do so. I would prefer NOT to give a list of textbooks at all. 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 practicing your programming – not by reading a textbook! I do NOT recommend spending any serious amount of money on textbooks!

With that health warning, here are the obligatory core texts for the module this year:

Jon Duckett: HTML & CSS: Design and Build Web Sites (Team Solo only)

 <u>http://www.amazon.co.uk/HTML-CSS-Design-Build-</u> <u>Sites/dp/1118008189/ref=la\_B001IR3Q7I\_1\_1?s=books&ie=UTF8&qid=14392</u> 0

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

 <u>http://www.amazon.co.uk/JavaScript-JQuery-Interactive-Front-end-</u> <u>Development/dp/1118531647/ref=la\_B001IR3Q7I\_1\_3?s=books&ie=UTF8&qid</u> =1439202657&sr=1-3

Lynn Beighley and Michael Morrison: Head First PHP and MySQL (Team Solo only)

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

Cay Horstmann: Big Java (late objects)

• <u>http://www.amazon.co.uk/Big-Java-Late-Objects-</u> Horstmann/dp/1118087887/ref=sr\_1\_1?s=books&ie=UTF8&qid=1442237249& <u>sr=1-1&keywords=big+java+late+objects</u>