

Module specification

1. Factual information			
Module title	TM105: Introduction to Programming	Level	1
Module tutor	TBA	Credit value	15 points
Module type	Taught	Notional learning hours	4 credit hours

2. Rationale for the module and its links with other modules

This module is an introductory level programming module and it is meant to provide basic foundation in computer programming to students. Students will learn how to develop solutions (algorithms) using pseudocode to solve simple problems. Thereafter, they will learn how to implement these solutions using a programming language (Java). This module serves as foundation for second level programming modules.

3. Aims of the module

The module aims to:

- Introduce the technique of solving simple problems using pseudocode.
- Introduce Java programming via writing, compiling and executing simple programs.
- Present how to store and deal with data including variables, constants, and expressions.
- Cover deeply the concepts of program control structure and illustrate each concept with a diagrammatic notation using UML.
- Present how these concepts are implemented in Java.
- Introduce the concept of modularization and how to write Java methods.
- Present how to deal with basic data structures like strings, arrays and two dimensional arrays.

4. Pre-requisite modules or specified entry requirements

Students are expected to have completed study of EL111 module before they can undertake this module.

5. Intended learning outcomes	
A. Knowledge and understanding	Learning and teaching strategy
<p>After studying the module, <u>the student will be able to:</u></p> <p>A1. Understanding of the design and programming processes</p> <p>A2. Knowledge of the main constructs and mechanisms in programming using Java language.</p> <p>A3. Understanding of the techniques used in developing a medium Java application.</p> <p>A4. Understanding of the basic data structures like strings, arrays and two dimensional arrays.</p>	<ul style="list-style-type: none"> • 25% face-to-face tutorial sessions • Video recorded sessions • TMA • MTA and final exam • Text book and supporting material

B. Cognitive skills	Learning and teaching strategy
<p>After studying the module, <u>the student should be able to:</u></p> <p>B1. Describe and apply key concepts and techniques in software design and development.</p> <p>B2. Analyse and abstract away from the details of a problem.</p> <p>B3. Design and formulate an appropriate solution to a problem and evaluate it.</p> <p>B4. Deal professionally with the basic data structures.</p>	<ul style="list-style-type: none"> • 25% face-to-face tutorial sessions • Video recorded sessions • TMA • MTA and final exam • Text book and supporting material

C. Practical and professional skills	Learning and teaching strategy
<p>After studying the module, <u>the student should be able to:</u></p> <p>C1. Create, develop and trace Java programs.</p> <p>C2. Use software tools such as a Java IDE and an On-line Java compiler.</p> <p>C3. Use appropriate programming skills.</p> <p>C4. Traverse data in the basic data structures in a professional way.</p>	<ul style="list-style-type: none"> • 25% face-to-face tutorial sessions • Video recorded sessions • TMA • MTA and final exam • Text book and supporting material

D Key transferable skills	Learning and teaching strategy
<p>After studying the module, <u>the student should be able to:</u></p> <p>D1. Find information from a range of sources to support a task.</p> <p>D2. Plan medium tasks.</p> <p>D3. Use Java libraries.</p> <p>D4. Use appropriate numerical, mathematical and abstraction skills.</p>	<ul style="list-style-type: none"> • 25% face-to-face tutorial sessions • Video recorded sessions • TMA • MTA and final exam • Text book and supporting material

6. Indicative content.

Chapter 1: Introduction to Computers, Programming, and Java
 Chapter 2: Elementary Programming
 Chapter 3: Selections
 Chapter 4: Mathematical Functions, Characters, and Strings
 Chapter 5: Loops
 Chapter 6: Methods
 Chapter 7: Single-Dimensional Arrays
 Chapter 8: Multidimensional Arrays

7. Assessment strategy, assessment methods and their relative weightings

TMA: 20% (Online Quiz 10%+ Small Project 10% (presentation 5% + code 5%))
 MTA: 30%
 Final Exam: 50%

8. Mapping of assessment tasks to learning outcomes

Assessment tasks	Learning outcomes															
	A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	C3	C4	D1	D2	D3	D4
TMA - Online Quiz	✓	✓	✓		✓	✓	✓		✓	✓	✓		✓	✓	✓	✓
TMA - Small Project	✓	✓	✓		✓	✓	✓		✓	✓	✓		✓	✓	✓	✓
MTA	✓	✓			✓	✓	✓		✓		✓				✓	✓
Final Exam	✓	✓		✓	✓	✓	✓	✓	✓		✓	✓			✓	✓

9. Teaching staff associated with the module
Name and contact details
TBA

10. Key reading list				
Author	Year	Title	Publisher	Location
Y. Liang	2014 10 th edition	Introduction to Java Programming, Comprehensive Version, Global Edition	Pearson	http://catalogue.pearsoned.co.uk/catalog/academic/product?ISBN=9781292070018
Paul Deitel and Harvey Deitel	2014 10 th edition	Java How to Program, Late Objects Version, Global Edition	Pearson	http://www.pearsoned.co.uk/bookshop/detail.asp?item=100000000574517
Cay S. Horstmann	2016 2 nd edition	Big Java, Late Objects	Wiley	http://eu.wiley.com/WileyCDA/WileyTitle/productCd-1119321077.html
Walter Savitch	2015 6 th edition	Absolute Java, Global Edition	Pearson	http://www.pearsoned.co.uk/bookshop/detail.asp?item=100000000613443
Tony Gaddis	2015 6 th edition	Starting Out with Java: From Control Structures through Objects, Global Edition	Pearson	http://www.pearsoned.co.uk/bookshop/detail.asp?item=100000000611758

11. Other indicative text (e.g. websites)
http://www.cs.armstrong.edu/liang/intro10e/ http://www.tutorialspoint.com/compile_java8_online.php