

Module 1: Computer Programming

Stage	1						
Semester	1						
Module Title	Computer Programming						
Module Number/Reference	1						
Module Status (Mandatory/Elective)	Mandatory						
Module ECTS credit	10						
Module NFQ level (only if applicable)	8						
Pre-requisite Module Titles	None						
Co-requisite Module Titles	None						
Is this a capstone module? (Yes or No)	No						
List of Module Teaching Personnel	Mr Eoin Carroll Mr John Hannon						
Contact Hours				Non-contact Hours			Total Effort (Hours)
Lecture	Practical	Tutorial	Seminar	Assignment	Placement	Independent work	
24	36			60		80	200
Allocation of Marks (Within the Module)							
	Continuous Assessment	Project	Practical	Final Examination	Total		
Percentage contribution	50%			50%	100%		

Intended Module Learning Outcomes

On successful completion of this module learners will be able to:

1. solve programming problems of modest complexity in a systematic, well-organised way
2. specify precisely the syntax and semantics of a programming language construct
3. document accurately the design of a program on-the-fly
4. determine the basic efficiency of an algorithm

5. design and develop a range of standard algorithms
6. analyse and modify a program written by someone else
7. develop a program using an integrated development environment

Module Objectives

This module teaches the learner how to design high-quality programs in a systematic way. All the relevant concepts and techniques are explained and exemplified in the clearest, simplest language. The module aims to introduce the learner to the concepts of programming and problem solving.

Module Curriculum

Introduction to problem solving

- How do you complete a task?
- Identifying sub-components on larger task
- Defining order of subcomponents
- Creation of algorithms
- Stepwise design of programs

Introduction to programming

- Expressions and statements
- Basic arithmetic
- Comments
- Variables and assignment (Integers, doubles, booleans, characters)
- Boolean expressions and logic
- Conditional statements
- Iteration statements
- User input
- Output
- String manipulation
- Sub-routines (Parameters, Signature, Procedures, Copy rule.)

Professional Practice

- Developing a good coding style
- Using comments effectively
- Naming conventions
- Indentation
- Code structure

Reading lists and other learning materials

Mullins, T. *A First Course in Programming with Java*, 2000.

Secondary reading

McGettrick, A D. Graded Problems in Computer Science, 1983

Eckel, B, *Thinking in Java 4th Edition*, Prentice Hall, 2006

Web Resources

<http://docs.oracle.com/javase/tutorial/tutorialLearningPaths.html>

Module Learning Environment

Accommodation

Lectures are carried out in classrooms / lecture halls in the College. Lab tutorials are carried out in computer labs throughout the Campus. All have the language software required to deliver the programme.

Library

All learners have access to an extensive range of physical and electronic (remotely accessible) library resources. The library monitors and updates its resources on an on-going basis, in line with the College's Library Acquisition Policy. Lecturers update reading lists for this course on an annual basis as is the norm with all courses run by Griffith College.

Module Teaching and Learning Strategy

The module is delivered through a combination of lectures, tutorials and practical lab programming sessions. The learners complete a series of worksheets throughout the module, which build on the learning in lectures. The emphasis is on developing practical programming skills based on sound theoretical knowledge.

Module Assessment Strategy

The module assessment consists of a series of continuous assignments and a final examination.

Element No	Weighting	Type	Description	Learning Outcome assessed
1	50%	Weekly Work Submission	A series of weekly worksheets covering the design and development of computer programs	1-7
2	50%	Closed Book Examination	End of module examination	1-6