

<b>Pre-Requisite Modules code(s)</b>	<b>Co-Requisite Modules code(s)</b>	<b>ECTS Credits</b>	<b>Module Code</b>	<b>Module Title</b>
<b>None</b>	<b>None</b>	<b>10</b>	<b>CMPU1025</b>	<b>Programming</b>

### 8.1.6. Programming

**Module author: Programming and Algorithms Group**

#### **Module Description:**

The module teaches the fundamental principles required to design, write, test and document structured procedural programs.

#### **Module aim**

The aim of this module is to

- Teach the fundamentals of procedural programming
- Teach the principles of good program design, implementation, documentation and testing.

#### **Learning Outcomes:**

On completion of this module, the learner will be able to

- Apply basic problem solving techniques to design a program using appropriate modules and data structures to specified requirements.
- Implement a program using previously a developed design
- Use an Integrated Development Environment (IDE) proficiently to develop programs
- To understand the compilation/link processes and interpret errors generated
- Design appropriate test data to ensure module and program correctness and robustness
- Debug a program using an IDE and by program tracing
- Write documentation for a program.

#### **Learning and Teaching Methods:**

This module will be taught using lectures, practical sessions in the laboratory and tutorials. All theoretical material will be taught in lecture class and this will include the rules and syntax of procedural programming. Each lecture will include many example programs to show the students how the material covered in the lecture is implemented.

Due to the practical nature of programming, a large emphasis will be placed on allowing the students to practice the development of programs in the laboratory. Each week, the students will be given a number of programming exercises that cover all material taught to them in their lecture class.

Tutorials will be used to allow the students to get extra tuition in the more difficult areas. They will also be used to allow the students to ask for any extra help required. The tutorial will incorporate

<b>Pre-Requisite Modules code(s)</b>	<b>Co-Requisite Modules code(s)</b>	<b>ECTS Credits</b>	<b>Module Code</b>	<b>Module Title</b>
<b>None</b>	<b>None</b>	<b>10</b>	<b>CMPU1025</b>	<b>Programming</b>

additional exercises and the ability for the supervisor to provide more one-to-one assistance with a student.

A Virtual Learning Environment (VLE ) is used extensively in this module.

### Module content:

- *Introduction:* What is a program? Source code. Machine code. Editing, Compiling Linking, Debugging. Use of an Integrated Development Environment (IDE). The command line.
- *Basic Data Types:* integer, floating-point and character data and variables.
- *Basic Input-Output:* Display data on a screen. Input data from the keyboard.
- *Programming Structures:* Conditional statements: Boolean values and expressions, logical and relational operators, if-statement, case-statement, compound conditional statements.
- Iterative constructs: while-statements, for-statements and nested control statements.
- *Introduction to Data Structures:* Strings, single-dimensional arrays, two-dimensional arrays, dynamically allocated arrays, user-defined structures, abstract data types, and enumerated data types. Command line arguments.
- *Structured Programming:* functions, parameter passing, returning values, global and local variables, nested functions, reusable code, library functions.
- *Implementing Basic Algorithms:* Summation, counting, numeric operations, swapping, maximum and minimum, simple string and array manipulation.
- *Testing and debugging:* Objectives and principles of testing. Choosing appropriate test data. Testing and debugging strategies. Debugging using an IDE. Debugging using a program trace.
- *Documentation:* Writing user and technical documentation. Style guidelines.

### Module Assessment

Assessment of the module is a combination of the following:

Continuous Assessment (50%):

- Individual assignments ,
- Lab tests,
- On-line tests
- In-class written tests

Written examination(50%): One three hour, end of module examination.

<b><i>Pre-Requisite Modules code(s)</i></b>	<b><i>Co-Requisite Modules code(s)</i></b>	<b><i>ECTS Credits</i></b>	<b><i>Module Code</i></b>	<b><i>Module Title</i></b>
<b><i>None</i></b>	<b><i>None</i></b>	<b><i>10</i></b>	<b><i>CMPU1025</i></b>	<b><i>Programming</i></b>

**Essential Reading:**

Depending on the procedural language used in this module, specific reading lists will be specified in advanced of the start of the module.

**Supplemental Reading:**

Dependent on the procedural language used.

**Web references, journals and other:**

Dependent on the procedural language used.

**Further Details:**

Class size is expected to be 80, broken into groups for tutorials and labs. Semesters: 2 Contact hours per week: Lectures: 2 hours Lab: 2 hours Tutorial: 1 hour

**Date of Academic Council approval.....**