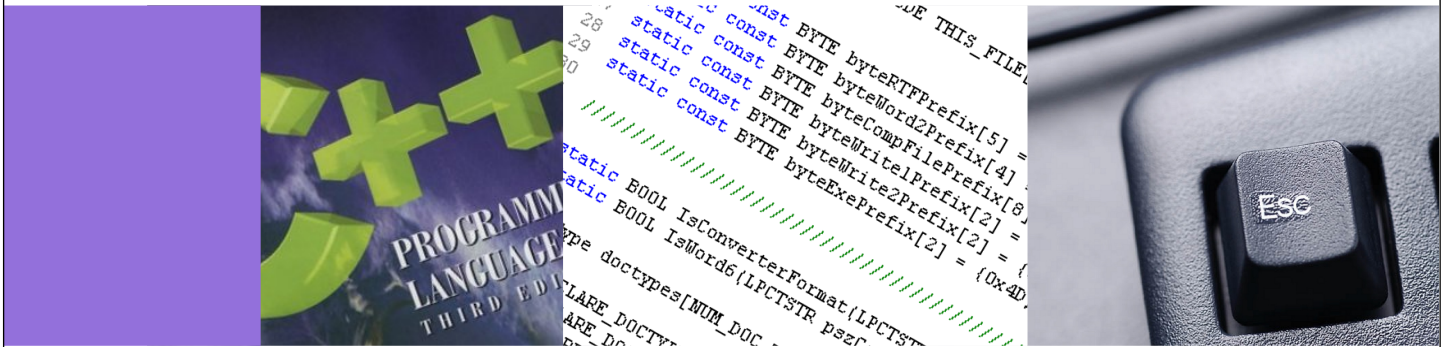# Unit MTH399 Advanced Mathematical Programming

This unit will deal with systematic ways to turn mathematical concepts and algorithms into computer programs, focussing mainly on the C++ programming language. We will emphasise defensive programming and unit testing as a means to produce rigorous and robust programs.

Lecturer: Doctor Andrew Burbanks
Department of Mathematics

The goal of this unit is to express mathematical concepts and algorithms as robust and efficient computer programs, using mainly the C++ programming language, together with some general programming concepts.

Starting with some very simple programs, we will build up expertise in using datatypes, writing functions, using the Standard Template Library, and defining simple classes.

C++ is a large and often difficult language that takes years to master. By taking a systematic approach, we cover some of the essentials, from a mathematical perspective, and aim to provide a strong base from which the student can gain future expertise.

Traditionally, many programmers would take the following approach: plan the program, write the program, and then spend many hours correcting mistakes (de-bugging). This is a painful process.

We will take a different, and more modern approach: we will begin by writing down some basic properties that our program should have, we will embody these in code in the form of "unit tests". Only when we have these tests will we write the program itself and ensure that the tests are passed. This and other "defensive programming" techniques will allow us to avoid many of the bugs that can otherwise occur.

Further information
Doctor Andrew Burbanks, Senior Lecturer
Department of Mathematics

Lion Gate Building, Room 1.48
andrew.burbanks@port.ac.uk
www.port.ac.uk/maths

University of Portsmouth