# CM2204: Advanced Programming Laboratory Worksheet (Week 7)

Prof. D. Marshall

## Aims and Objectives

After working through this worksheet you should be familiar with:

- Be able to use input/output streams in C++;
- Be able to read and write text files in C++;
- Be familiar with the `string` and `vector` classes and their C equivalents;
- Understand the terms *operator overloading* and *references*.
- Understand the difference between `struct`s in C and `struct`s and `class`es C++;
- Be able to implement a simple class with member functions in C++;
- Use `public` and `private` to hide implementation;

**None of the work here is part of the assessed coursework for this module.**

- Follow the web links for files highlighted and underscored to get code listings
- All lecture and lab class code is a available on the CM2204 Web page
- Solutions to the exercises will be released on the CM2204 Web page in **Week 8**.

# C++ Basics

1. Write a C++ program which prints out a table of Fahrenheit temperatures and their Celsius (centigrade) equivalent. You may recall that the formulae for conversion are:

   $$F = \frac{9}{5}C + 32$$

   $$C = \frac{5}{8}(F - 32)$$

   Use `cout` and the $<<$ operator for output.

   - Modify your answer so that it *prompts* the user to enter the number of decimal places that should be used in the output —- use *stream manipulators* to set the precision and align the output. *What does the fixed I/O manipulator do?*
   - Create a C++ and header file which contains a function `fahrToCel` which performs this conversion, and call it from a main function in a separate file. Add the function to a namespace `numconv`.

2. (Question 5 of chapter 2 of Thinking in C++) Change Fillvector.cpp so that it prints the lines (backwards) from last to first.

   - Modify the code so that it writes the output to a second file.

3. (Question 7 of chapter 2 of Thinking in C++) Write a C++ program that reads in a text file and displays the file a line at a time, waiting for the user to press the Enter key after each line.

4. (Questions 4 and 5 of Chapter 4 of Thinking in C++) Create a struct with a single `int` data member, and **two** global functions, each of which takes a *pointer* to that `struct`.

   - The *first function* has a second `int` argument and sets the `structs` `int` to the argument value,
   - The *second function* displays the `int` from the `struct`.

   Test the functions.

   - Now **move** the functions so they are *member functions* of the struct, and test again.

5. Write a C++ class which represents a *circle*, with **private** member variables to store the *radius* and the value of $\pi$ (use $\pi = 3.1419$). Add **public** functions to:

   - **obtain** the **value** of the *radius*,
   - **set** the **value** of the *radius*, and
   - **calculate** the *circumference* of the circle given by $2\pi r$, where $r$ is the *radius*.

## Further Practice

1. Compile and test the Stack example from Chapter 4 of Thinking in C++.

2. (Advanced) Question 10 of Chapter 4 of Thinking in C++.

3. Question 2 of Chapter 5 of Thinking in C++.