# Algorithms and Data Structures 2

## Laboratory Sheet 1

You have a choice of development environments, check out the *Syntax References and IDE's* link in Brightspace. If you are using the online compiler, when you are finished with a program, you can cut and paste it into a notepad file and save it with a .c extension onto your OneDrive account or hard drive (or you can create an account on *repl* and save to a repository in GitHub).

1. Write a Hello World program and satisfy yourself you know how to run and save it.

2. Write a program which accepts a temperature in degrees Fahrenheit and converts it to degrees Centigrade, displaying the answer. The formula for converting a Fahrenheit temperature to Centigrade is:

   Centigrade = (Fahrenheit - 32.0) * (5.0/9.0)

3. Write a program which asks the user to input 3 integers and then calculates and displays the average of the integers as a real number. Note what answer you get when you input values 3, 5 and 6 and the average is 4.67

4. Write a program called to input a person's first initial, their age and their height (a real number). Then display the values using following illustration as an example:

   ```
   Personal Statistics
   ==================

   Identifier      Age          Height
   T               19           6 Feet
   ```

   Once this is working, experiment with omitting the & from the *scanf* statements and inserting it into the *printf* statements to see what results you get.

5. In a triangle, the sum of any two sides must be greater than the third side. Write a program called **triangle.cpp** which accepts three values representing the sides of a triangle and determines whether they form a valid triangle. If they do form a valid triangle, determine if that triangle is **equilateral**, **isosceles** or **scalene**.

   This program will require the use of an *If* statement and logical operators (and, or, not) which all have the same structure and syntax as Java.

5. Using the program in the first lecture (go lecture 1, slide 46) as a guide, write a program called **serviceCharge.c** to input 5 account balances for customers. For each account balance, check to see if a service charge needs to be applied. If the account balance is 100 Euros overdrawn (i.e. balance is exactly -100), then a service charge of 10 Euros is applied (i.e. subtract this amount from the balance). If the account balance is over 100 Euros overdrawn but less than 500 Euros, the service charge is 100 Euros and for figures greater than 500 Euros overdrawn, the service charge is twice the deficit (e.g. if they are 700 euros overdrawn, the service charge is 1400 euros). No service charge applies if the account is not overdrawn. The program should input and display each of the 5 balances in turn with the service charges that were applied.