Discipline of Computing and IT
University of Newcastle

SENG1120/6120 – Semester 1, 2018
Lab 3 (Week 3)

In this lab, you will practice the creation of overloaded operators and inheritance.

Note: `#include <cmath>` gives you access to mathematical functions such as `sqrt()` and `pow()`.

1. Create class `Point` that stores the x and y coordinates (stored as `double`) of a point in the *Cartesian plane*. It should be possible to create instances of `Point` with either default values representing the origin, i.e. (0,0) or with user-provided coordinates. The class should also provide mutating member functions `set_x()` and `set_y()` that allow separate setting of the x-coordinate and the y-coordinate. The class should provide query functions `get_x()` and `get_y()` that allow the x-coordinate or the y-coordinate to be retrieved. You should also overload the `cout <<` operator to output `Point` using the notation (x, y). Demonstrate the behaviour of your new class with a demo file.

2. Define a function `length()` that takes as parameters two instances of `Point` and returns the length of the line joining the points. Demonstrate the behaviour of your new function.

3. Define the comparison operators `==` and `!=` when applied to a pair of instances of class `Point`. Demonstrate the behaviour of your overloaded comparison operators.

4. Define the operator += as applied to instances of class `Point`. This member function will have the effect that, if `A` and `B` are instances of class `Point`, then `A += B` yields the same result as `A = A + B`.

5. Define a class `Point3D` that **inherits** `Point` and adds a third coordinate, z (this class can be used to represent points in the *Cartesian space*. This task will require some methods to be re-written. Demonstrate steps 1-4 with the new class.

Good Luck