SCSSE, University of Wollongong
CSCI204/MCS9204 Object and Generic Programming in C++
Laboratory Exercise week 4

Task 1: Debugging

In lab materials you will find two programs **Student.cpp** and **Cake.cpp**. The programs have various errors; syntax, logical and/or with respect to the sample output. Copy the files to your own working directory and fix the bugs.
The correct output for **Student.cpp** should be:

```
Student a is year 2
Student b is year 1
Student c is year 3
Student a is now year 3
Student b is now year 2
Student c is now year 4
```

The correct output for **Cake.cpp** should be:

```
Cakes are white with butter cream icing and 1 candle unless otherwise indicated
Order #111 chocolate cake with whipped cream icing and 8 candles
Order #222 yellow cake with chocolate icing and 1 candle
Order #333 banana cake with butter cream icing and 1 candle
Order #444 chocolate cake with butter cream icing and 1 candle
```

Task 2: Constructors and a friend functions

Write code **Customer.cpp** with two classes defined:

1. The first, `Customer`, holds customer data, specifically an identification number and a postal code.
2. The second, `City`, holds city information, specifically the city name, state and postal code.
3. Each class should have a constructor with parameters to set the relevant values for the data members.
4. Create friend functions that display a customer's number, city name, state and postal code.

Write a short main() function to test the constructors, the member functions, and friend functions.

Task 3: Composition

Make a copy of the solution to Task 3, call it **Customer2.cpp**, and modify it to perform the following:

1. Put a `City` object inside the `Customer`, with the postcode for the `Customer` just in the `City` object.

2. The `Customer` constructor should call the `City` constructor.
3. Write `display` functions in `Customer` and `City` to use instead of the friend functions.
4. In the `main()` should be something like ...
```
Customer aCust(1572,"Wollongong","NSW","2500");
aCust.display();
```
5. Try and access the postcode field directly from your `Customer::display()` Can you?

Task 4: Destructor

Write code in **MyClass.cpp** with a class definition for the class **MyClass**. The class should have an integer data member for the number of elements, an integer pointer, two constructors (default; with parameters)and a destructor. One of the constructors should have two parameters. It should allocate dynamic memory for the integer pointer according to the number of integers and assign values from an integer array that passed into the constructor to the private array that the pointer points to. Display the integers of the dynamic array. Define a destructor to release the dynamic array.

Provide an example of the main() function to demonstrate the instantiation of one `MyClass` object with initial values.