Practical 2 (due 2022-08-05 @ 09:00)

The purpose of this practical is for you to become familiar with the basics of C++ classes.

A Portable Gray Map (PGM) is a Netpbm grayscale image format. It is different from PPM in that each pixel is represented by one integer (instead of three for PPM).

- Create a data structure to represent a **`Canvas2D`**, which manages a dynamic two-dimensional array of integers, where each integer represents a pixel of a PGM image. The **`Canvas2D`** data structure must contain the following:
  - The number of rows (height) / columns (width) of the PGM image
  - A two-dimensional dynamic array of pixels representing the PGM image
  - A set of constructors for:
    - Creating a default **`Canvas2D`** (using associated class constants)
    - Creating a **`Canvas2D`** where the dimensions of the PGM image are specified
    - Creating **`Canvas2D`** where the dimensions of the PGM image and a default value for all the pixels are specified
    - Creating a copy of an existing **`Canvas2D`**
  - A set of accessor member functions for both the rows and columns as well as to allow access to each pixel
  - A mutator member function for changing the value of a pixel by specifying its row and column. The size of the two-dimensional array should not be changeable
  - A destructor
  - A set of utility functions for ensuring that values related to the size of the **`Canvas2D`** are in range
  - A `toPGM` method that creates and returns a PGM string

The creation of a UML class diagram design will be covered in both class and the tutorials next week.

Create a suitable `main` function which demonstrates the usage of the class. The `main` function must test the data structure by drawing a face similar to the one shown in Figure 1 in a `main.cpp` file (Hint: you will have to create a method for drawing a circle and another for drawing a rectangle in **`Canvas2D`**, which you will use as you draw the face in `main`).

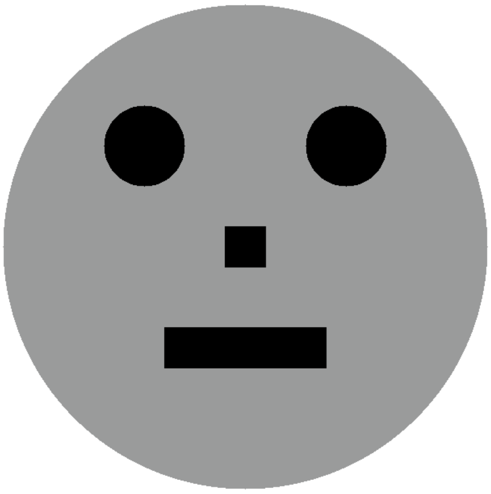

Figure 1: The image to be drawn in `main`.

	Mark sheet	
	Design (to be discussed in class on this coming Monday)	10
	Class declaration / definition (in .h and .cpp file)	10
	Declaration of Canvas2D structure	10
	Constructors	10
	Copy constructor	10
	Destructor	10
	Accessor member functions	10
	Mutator member functions	10
	Utility member functions	10
	Demonstration of class functionality in a main function	10
	Total	/100