

# Programming Assignment 1

## Image Processing Framework Application Creation

### Objective:

In this assignment, you will create a GUI application that performs the following tasks - loading an image, displaying an image, saving out an image, performing some interpolation tasks.

### Tools

*Image Files* : A set of standard image database for use in this course is provided on Moodle course page. The file format that we will work throughout the course is the Portable Pixel Map (PPM) file format. For details refer the website <http://netpbm.sourceforge.net/doc/ppm.html> . In the context of this course, a physical file refers to one image. Hence, we will use the plain PPM format (described at the bottom of the above webpage). Note that the pixel information is stored as Red, Green and Blue channels. Each channel should be stored and operated on independently.

### GUI Application :

As an example, a sample GUI created with the **wxWidgets** library (<http://www.wxwidgets.org/>) is provided. Download the latest stable version of the library, compile it on your operating system of choice, and create the static / dynamic libraries for use in the programming projects. Follow the installation and compilation instructions provided on the Moodle page under “Required Software”. It is recommended that you also compile and run their “samples” (in the “samples” subfolder).

Refer to the following resources for information about wxWidgets:

Tutorials (<http://www.wxwidgets.org/docs/tutorials.htm>)

wxWiki ([http://wiki.wxwidgets.org/Main\\_Page](http://wiki.wxwidgets.org/Main_Page))

Forums (<http://forums.wxwidgets.org/>)

The “Tutorials” page is highly recommended for learning the ropes in wxWidgets world. An accompanying document “How-to compile wxWidgets on Visual Studio 2010” describes how to download and compile the library on MS Windows. Similar procedures exist for Linux and Mac OS X distributions as well.

**NOTE:** You can use your own GUI library that can be compiled and run in any digipen PC. Your submission should be able to compile and run in any digipen PC without changing any settings/configurations in Visual studio environment.

### Grade Distribution :

- (a) Loading an image file specified by the user correctly (no crashes) - **(5 marks)**
- (b) Saving the currently loaded file to disk using valid PPM format specification - **(5 marks)**
- (c) Displaying the image on screen - **(10 marks)**

The image should be scaled to match the client window size at all times. If the user resizes the window manually, the image should be resized accordingly. Use the following methods of interpolation - (a) Nearest Neighbor, and (b) Bilinear Interpolation. **(5 marks each)** for implementing the method correctly

All the above operations should be **menu driven**.

**Deliverable:** For submitting your assignment, create an archive that is named as follows:

**CS370\_<student\_login\_name>\_<assignment#>.zip**

Example: If your login is pacquiao.mandy and assignment 1 is being submitted, your folder would be named **CS370\_pacquiao.mandy\_1**.

**Include the following in the archive:**

- (a) A folder named **src** – It contains complete source code of the application including any project/solution files that you create. The instructor will not be responsible for creation of Makefiles or VS Projects/Solutions to make your project work. Instructor will compile your code and if your code does not compile to produce an executable, it will not be graded.
- (b) A folder named **doc** - It contains README.docx having a short description of your solution to the programming project. This should include a description of your data structures for loading and storing images. Include screenshots of your application in operation. Download README.txt from the Moddle course page. You should submit the document as PDF file. If this document is not in proper format, or the sample output images are not from your implementations, then you will get ZERO grades for this assignment.
- (c) A folder named **release** – It contains your executable. The instructor should be able to run this executable.

Your project should compile and execute without errors or warnings in the latest Visual Studio environment on Digipen computers.

**Failure to submit the deliverable in the required folder format will result in a zero assignment grade.**

**Late Submission:** Refer to the course outline for the late submission guidelines.

**Hints for implementation:**

1. Read the tutorials at <http://www.wxwidgets.org/docs/tutorials.htm>. There are easy step-by-step instructions on the page for building applications using wxWidgets.
2. In particular, read the article “[What Do These Sizer Things Do?](#)”.
3. Use the tutorials on the page “[Intermediate wxWidgets Tutorial](#)” to learn how most common widgets are initialized and then used.

### Grade Sheet for Assignment-1:

Criteria	Points Possible
Loading an image	5
Saving an image	5
Displaying the image	10
Nearest neighbor interpolation	5
Bilinear interpolation	5
<b>Total</b>	<b>30</b>

#### Important Note:

Always rebuild the solution whenever there is modification in the code. The instructor should be able to rebuild/execute your project on any computer at DigiPen having the following software installed:

1. Visual Studio Community 2022
2. wxWidgets-3.0.5