國立臺北科技大學 自動化所 – 人機介面

API Design –Midterm Project

Starts from Friday, 4/14 Due Friday, 4/21

Objective

The purpose of this project is to have the experience of creating software for image processing. The main objective of this project is to create the core APIs of image processing that actively communicates C# GUI being a MDI viewer.

Materials

Before you start this project, you will need to download Visual C++ IDE and the source codes from NTUT Tech. i School. It is critical that you get the source codes downloaded and functioning as early as possible in the week so that if you have problems you will have a chance to talk me and get help. If you wait until the Monday (or later!) and you cannot get the source codes downloaded and tested, it is unlikely you'll finish this project on time.

Required Tasks

1. Follow the instructions (separate document) to design and run the "CImage" API with "CVIP" project. Do not proceed to the next steps unless your API functions as expected and execute without errors. (50%)
2. Design the flat C API of the image processing algorithms extracted from "CVIP" project (Choose any two algorithms of them) with your "CImage" API. The instructions document will be given for this part. Please ensure this is executable. (50%)
3. Avoiding the problems listed in the Evaluation section below is part of the required tasks of every assignment. This list grows as the semester progresses, so be sure to check it again with each assignment.

Evaluation

In all of the assignments this semester, designing quality API that executes without errors and then testing the resulting application and iterating until it functions properly is the goal.

Here are the most common reasons assignments are marked down:

- Project does not build.
- Project does not execute without errors.
- One or more items in the Required Tasks section were not satisfied.
- A fundamental concept was not understood.
- Code is sloppy and hard to read (e.g. indentation is not consistent, etc.).
- Your solution is difficult (or impossible) for someone reading the API to understand due to lack of comments, poor variable/method names, poor solution structure, etc.
- Assignment was turned in late.

Often students ask “how much commenting of my API do I need to do?” The answer is that your API must be easily and completely understandable by anyone using it. You can assume that the reader knows C++, but should not assume that they already know the (or a) solution API to the problem.

Submission (Deadline: 2017/4/21 13:00)

1. Source code of your design for the tasks (including .exe/ .h / .cpp /.cs / .dll files)
2. Documents for describing your design and how it works