EECS 22L: Project 2 Grading Criteria

Prepared by: Weiwei Chen, Che-Wei Chang, and Prof. Rainer Dömer

March 5, 2013

The follow items are mandatory for the *beta* version of the *OCR* program:

1. For the Beta release, we are expecting an OCR program which is capable of transforming the clean colorful image of the HelloWorld program into a compilable C program source code file.

   The input sample source images are:
   `eecs22/ocr_scans/30a_ColorWorld_Clean300DPI.jpg`,
   `eecs22/ocr_scans/30b_ColorWorld_Clean200DPI.jpg`,
   `eecs22/ocr_scans/30c_ColorWorld_Clean100DPI.jpg`.
   You can use any one of them as the input for your OCR program.

2. To achieve the first item, the following OCR functions may be desirable in your program. However, it depends on your design whether to implement any of the following functions or not.

   - Support for file I/Os, such as load an image file to the program, save the source code file on the disk drive, save the processed images, etc.
   - Support for digital image processing, such as turning the colorful input image into black and white; increasing the contrast, sharpness, and focus of the image; image cropping for a single character; image rotation for leveling; and so on.
   - Basic infrastructure for the character matching database, e.g. create / delete / clean the database, add an entry to the database, fill the content of the database with letters, digits, symbols, etc.
   - AI support for character image pattern matching and optical character recognition
   - Support for input variations, such as different fonts and images with different DPIs
   - A user interface for the program
   - Unit test for different modules
   - Support for vocabulary / word matching with the database for speed pattern matching and spell checking

   Note: We are expecting a program which can accomplish the first OCR requirement. Please state clearly in the `README` file as well as in the OCR user manual, i.e. `man/OCR.pdf` (preferably with demonstration snapshots) about how to use your program to transform an input image of C source code in a compilable program file.

3. A well structured project folder with **compilable** source code files and corresponding documentations are also mandatory for the beta version. Specifically, we are looking for:

   - A complete and clean tarball with all the project files in proper directories
   - Proper project file hierarchy (as what was presented in week3's discussion session)
   - An `INSTALL` file with the descriptions of two installation options:
     - tarball extraction
     - CVS checkout information, i.e. linux command that can be used to get the project checkout

     Proper instructions on how to install the program.

- A README file with the information of the authors, program version, date, and general information / description about the software.
  Note: for the alpha version, please also state the basic functions that have been implemented in your program and how to use / test them. This is critical to the grading.
- A COPYRIGHT file with authors and copyright information
- A working top-level Makefile with at least three targets, i.e. 'all', 'test', and 'clean'.
  Note: this file should be different from the Makefile in the *src* directory.
- The src directory with all the properly documented program source code files.
- The bin directory with the binary executable file of the OCR program.
- The doc directory with all the documentation files for this project, i.e. OCR_SW_Spec.pdf
- **New for beta**: A pdf file named OCR.pdf in the doc directory as the user manual of the *OCR* program. Screenshots of the program functions are desirable to have in this document
- **New for beta**: An ASCII text file named OCR.1 in directory man/cat1/ as the static text file for the *OCR* program's manual page
- The test directory with the unit test modules, a Makefile for unit testing and, instructions on how to do the unit testing. **Note:** it is desirable to use the 'make test' target in the top-level makefile to execute all the test targets in the Makefile in the test directory.

Good luck and Happy Coding!