PRIM Project - Small Graphic Language

User Manual

Richard CHEAM

1 Tutorial

!!ATTENTION!! to understand how it works please read EVERYTHING through

if something went wrong, please kindly execute the program manually using the line provided below (they will be in another text file as well namely '*exec.txt*' so that it is more convenient for the user to copy and paste).

Note: to function the SDL library, the directory that contains it will be needed to be in the same folder where the program is.
Note: if the input file, for example, *virus.ipi* is not exist in the directory, the execution will display segmentation fault. Be sure, the input file is in the same directory and beware with name of the file as well in case it is different when being execute.

Makefile is also provided in the folder, but just in case it does not work as the way I plan. And also the **Makefile** is only for **prog NOT prog_sdl**, so by using **Makefile**, the program will not display the graphical interface.

START HERE

- In terminal, proceed yourself into the directory which contains the program.
- Then, there are two executable files **prog** and **prog_sdl**. **prog_sdl** will be the one with graphical interface. (I divided it into two just in case the **prog_sdl** run-time takes too long, but it works fine)

  Three ways to input the *ipi* file by using **prog** are:

  $$./prog < test.ipi$$

  OR

  $$./prog \ \ test.ipi$$

  OR

  $$./prog \ \ test.ipi \ \ image\_name$$

  Then

  $$display \ \ image\_name.ppm$$

  Three other ways to input the *ipi* file by using **prog_sdl** are:

$$./prog\_sdl < test.ipi$$

OR

$$./prog\_sdl \;\; test.ipi$$

OR

$$./prog\_sdl \;\; test.ipi \;\; image\_name$$

Then

$$display \;\; image\_name.ppm$$

Note: if you happen to use the first two methods to run the program, the output image's name would be '**richard_img.ppm**'; else, it will be what you typed after the *ipi* file. Plus, you do not need to type the extension **(.ppm)** at the end of the image_name, the program will handle it for you just like the instance above.

- Lastly, if the user is intended to see the **graphical interface**, the program will pop out a window called **"WIN"** and start showing the construction of the image. Then, after the illustration finished the user has to close **"WIN"**, otherwise the program will still be running in the terminal.
- The final product in **ppm** extension either '**richard_img.ppm**' OR '**your input after *ipi* file**' will then be produced and can be viewed.

Command to execute directly:

for **Linux** user:

$$gcc -Wall \;\; -o \;\; prog\_sdl \;\; main\_sdl.c \;\; function.c \;\; list.c \;\; stack.c \;\; `pkg-config \;\; --cflags \;\; --libs \;\; sdl2` \;\; -lm$$

for **Mac** user:

$$gcc \; -Wall \; -o \; prog\_sdl \;\; main\_sdl.c \;\; function.c \;\; list.c \;\; stack.c \;\; -I \;\; include \;\; -L \;\; lib \;\; -l \;\; SDL2 \; -lm$$