Lab 10 Option 2:
SDL Tutorials - Simple Drawing UI
due: April 29, 2016 11:59pm

Important Notes

- Submission location (Mercurial repository): /rd/cs246s2016/$USER/lab10
- This is a hard deadline. No late submission is allowed.
- **This assignment can be done in pairs or groups** If you need help, see the instructor or TA.
- Read through this specification completely before you start.
- Some aspects of this specification are subject to change, in response to issues detected by students or the course staff.

The goal of this lab is to make a simple drawing program that can draw ellipses and rectangles. For lab 09, if you are did the GUI option, then this lab is for that option. If you did the Baby Names Part II Option, then you MUST do option 1 for this lab.

Notes:

1. You may use EasyGraphics.h, EasyGraphics.cpp, and EasyColors.h as a starting point for this lab.
2. You may also use code from Lab 09 for this lab.
3. You will include the libraries with -l SDL, -l SDLmain, and -l SDLgfx using a Makefile
4. you will be swapping the location of objects in the collection that they are in. Consider what the best data structure will be for quick access and easy swapping of drawn objects/commands.

Tasks:

1. Add code that uses mouse down, mouse dragged, and mouse up events to give your program the starting and ending location of a drawn element.
2. Create a button to change the mode to draw ovals.
3. Create a button to change the mode to draw rectangles.
4. Create a button so that you can click a shape to send it one level back in the drawing sequence (drawn one shape later).
5. Create a button so that you can click a shape to send it one level forward in the drawing sequence (drawn one shape earlier).
6. Create a button so that you can click a shape to send it to the back of the drawing sequence (first thing drawn).

7. Create a button so that you can click a shape to send it to the front of the drawing sequence (last thing drawn).
8. Create a button so that you can click a shape to remove it from the drawing (Only the highest object that the mouse is within should be removed).
9. Create a button so that you can select a shape for editing.
10. Create a button that allows you to select a fill color for then next item drawn or for the currently selected object.
11. Create a button that allows you to select a border color for then next item drawn or for the currently selected object.
12. Create a print button, that prints a list of processing commands to draw the picture programmatically.

What to hand in: All of your source code files, a makefile, and a Readme.txt describing what you did.