Hashir Safdar
Project Overview

I will create a simulation of a canvas and a paintbrush named Create for my final project. Many simulations of this real life activity exist, such as the Windows program Paint. My program will load up any image, or a blank canvas, and then allow the user to use the mouse to edit the canvas in a variety of ways.

1. user loads up program, choosing to load an image or a blank screen onto a window
2. user selects a brush from a set of different brushes
3. using the mouse and start and release clicks, the user performs the brush's actions on the window

Potential list of brushes (customize thickness for each):
basic paintbrush - select a color and draw
highlight - lightens colors
darken - darkens colors
invert - inverts colors

Classes & Functions

Create() - main class to hold all objects, and construct a window.

loadImage(fileName) - loads image from .gif file and creates a window, returns window

loadBlank(width, height, color) - loads blank canvas, returns window

main() - creates window based on user terminal input, also assigns brush based on user input on the terminal.

has an infinite while loop that quits when a certain input is given.

rotate(window) - potential extra function that can rotate an image clockwise, called from main(), returns window

Brush(window, brushType) - brush class to deal with mouse input/output and pixel manipulation.

instance variables: currentBrush - string with what mode brush is currently in
window - passed in from Create()
brushType - type of active brush

draw() - looping and waiting for user mouse input.

uses clicks to determine where user is interacting and calls appropriate drawing function.

ends when a certain key is pressed, returning back to main() logic.

drawBrush(color, thickness) - pixel manipulation for basic brush, returns modified window

drawDarken(thickness) - pixel manipulation for darken brush, returns modified window

drawLighten(thickness) - pixel manipulation for lighten brush, returns modified window

drawX - for potential other implementations of brushes

BrushPanel() - potential extra window with images of tools for user to click on instead of terminal input

    getInput() - returns what brush user chooses to Create

Plan

I will start by implementing the mouse input/output part of the draw() function, testing on a blank window. I will then test basic pixel manipulation with drawBrush(). Following this, I will implement the rest of the brushes, and end with moving the test code into Create and implementing the ability to load an external image into the window.

Extras

If possible, I will add more abilities to modify the image, including common image editing features such as cropping/rotating, and potentially even combining images. I would also like for the ability to choose what brush the user wishes to use with more than just the terminal, including the mouse and keyboard.