Assignment 01

CSE 241

March 14, 2025

You are to make a C++ program which opens a window. Within the window is a coloured rectangle which changes its shade each time the mouse is clicked.

Description

Use of SDL2

To open a window, draw a rectangle, and respond to events such as mouse clicks, it is necessary to use an external library to do it. In this course, we use SDL2[1].

Any C++ file which makes use of SDL2's library[2] should begin with:

[1] It's also what the TAs have installed to test your submission with!

[2] Such as your assignment submission!

```
#include <SDL2/SDL.h>
```

To open a window, it suffices for this assignment[3] to do:

[3] In future assginments, we will checked the returned values to ensure no errors occurred

```
SDL_Init(SDL_INIT_EVERYTHING);
SDL_Window* window = SDL_CreateWindow("Assignment 01",
     SDL_WINDOWPOS_CENTERED, SDL_WINDOWPOS_CENTERED, 800, 600,
     SDL_WINDOW_SHOWN);
SDL_Renderer* renderer = SDL_CreateRenderer(window, -1,
     SDL_RENDERER_ACCELERATED);
```

It is not necessary to understand what all of these initialization function calls do at this time. If you wish to understand better:

- Calling `SDL_Init` is required before using any other SDL2 functions.
- `SDL_CreateWindow`, as its name suggests, creates a new window.[4]
- Introducing the `renderer` variable wih the `SDL_CreateRenderer` function is necessary for all future drawing operations, and will cause drawing to take place in the created window

[4] In this case, our window is 800 pixels wide and 600 pixels high

In your `main` function, you should set up a loop to repeatedly call the `SDL_PollEvent` function, like so:

```
SDL_Event event;
while (SDL_PollEvent(&event)) {
   // ... code to handle the event
}
```

There are two events that need to be handled, by checking `event.type`:

`SDL_QUIT` — Triggered when the user closes the window. In this case, your program should terminate.

`SDL_MOUSEBUTTONDOWN` — Triggired when the user clicks somewhere in the window. In this case, you should make the colour of the window slightly darker.

The colour of the window can be set with:

```
SDL_SetRenderDrawColor(renderer, r, g, b, 255);
SDL_RenderClear(renderer);
SDL_RenderPresent(renderer);
```

This will set the colour of the entire window to r, g, b (red, green, blue), where each r, g, b value is an integer between 0 (dark) and 255 (bright).

Initially, the colour of the window should be set to 255, 0, 127, which is a reddish-purple. After clicking once, the colour of the window should be changed to 247, 0, 123. Each time the user clicks[5], the red value should be decreased by 8, and the blue value should be decreased by 8. If the user clicks enough times to make the window black, it should remain black after that.[6]

[5] anywhere in the window

[6] The r, g and b values should never become negative.

When the user chooses to quit the program, the following sequence will correctly shut down SDL2:

```
SDL_DestroyRenderer(renderer);
SDL_DestroyWindow(window);
SDL_Quit();
```

Requirements

These are requirements for the C++ code:

- The window must open immediately when the program is run
- The entire window must be coloured with the same colour (initially reddish-purple)
- Each click should decrease the red value by 8 and the blue value by 4
- It should not be possible to have an r, g or b value outside of 0–255
- The program should exit when the user attempts to close the window
- There should not be any memory errors or crashes when running the program[7]

[7] On some platforms, SDL2 may have a memory leak when the user types in keystrokes. We will ignore this.

Submission requirements

You are required to submit a .zip file which contains two files in it:

`a01.cc` — The C++ source file for the assignment, as described above

`CMakeLists.txt` — A CMake build file. I would recommend using the CMakeLists.txt provided on Blackboard as a starting point. It must create an executable file call `a01`.