

# CMSI 371-01

## COMPUTER GRAPHICS

Spring 2016

### Assignment 0204 Feedback

Outcomes that eventually cover both 2D and 3D max out at | for now because we are dealing only with 2D in this assignment. They will expand to their full potential with the 3D course work.

Dustin Kane

ranneyd / dustinpkane@gmail.com

*Notes while running (high-priority notes are marked with \*\*\*):*

- Generally just fine, though now that I have seen the sprites and what you're doing with them, I don't think the animations being done are as "incompatible" with the keyframe perspective as I had originally thought. We'll see what comes out in the code. (+1a, +3a)
- Animation buttons are helpful in understanding your perspective on animating the sprites. (+4a)
- No runtime issues or problems seen. (+4a)

*Code review (refer to <http://lmucs.github.io/backing-guidelines/> for code-review abbreviations):*

1. <http://lmucs.github.io/backing-guidelines/javascript/#js-var> (4b) —Applicable in multiple places.
2. Having now seen your code, I can concretely say that the core difference between the way you were thinking of animation vs. the keyframed way is your association of the movement with an "uber-index" whose progression affected a variety of properties concurrently. As our discussions showed, this is still technically correct animation, but just not the way I've known animators to think. Instead, all of these properties are autonomously changeable, and if they happen to go in step with each other every now and then, that is more incidental than intrinsic. We'll see how your code shifts with the next assignment. But the pieces are definitely in the right place; it's a matter of how they are put together.
3. In your explosion sprite, your design choice of having a helper `drawExplosion` function is a good call. The one shift in thinking that I would suggest is to observe that your `w` and `h` parameters are really ultimately scale factors, and so you can reduce your helper function to just the desired color. The various widths and heights would be set by invoking `scale` before each `drawExplosion` call. (4b)
4. The way your sprites connect to the canvas via an ID is a decent level of indirection but ultimately still somewhat limited. What if the desired canvas has no ID? What if offscreen drawing is being performed? In the end, passing a rendering context draws the cleanest line of separation, I think. (4b)

1a — +

2a (max |) — |

3a (max |) — |

4a — +

4b — | ...Just reflects the individual small hiccups spotted relating to this outcome.

4c — +

4d — +

4e — + ...Descriptive messages and excellent frequency & timing.

4f — + ...Submitted on time.