

CMSI 371-01

COMPUTER GRAPHICS

Spring 2016

Assignment 0225 Feedback

Outcomes that eventually cover both 2D and 3D continue to max out at | for now because this assignment remains in 2D.

Christopher Dellomes

cjdellomes / cjdellomes@gmail.com

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

- Fun cartoon! The improvements in the sprites are readily apparent (that is still pending re-evaluation).
- No technical issues encountered at runtime, so we can just go to the code...

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

1. Custom property code spotted and fairly straightforward. Nice logic there for differentiating tweenable vs. non-tweenable properties. (+2a, +3a, +3b)
2. Custom easing functions seen, but the functions are not strictly non-monotonic. Unfortunately this diverges from the assignment instructions. (3b, 4a, 4d)
3. Clean up: square and circle aren't used anymore, so they should go. (4b, 4c)
4. In *keyframe-tweening-demo.js*: <http://lmucs.github.io/hacking-guidelines/javascript/#js-eq> (fortunately you do use `===` in *keyframe-tweener.js*) (4b)

1a — +

2a (max |) — |

3a (max |) — |

3b (max |) — |

4a — +

4b — | ...Vestigial code; use of `===`.

4c — | ...Though vestigial code is relatively harmless, we leave this here as a reminder that it is a good idea to clean it up aggressively.

4d — | ...Missed the request for non-monotonic easing functions.

4e — Good frequency and descriptive messages, with work dating from before the original due date (we'll count your sprite revisions here). (+)

4f — Submitted on time. (+)