CMSI 371-01

COMPUTER GRAPHICS

Spring 2013

Assignment 0129 Feedback

Because this assignment covers a beginner-level exercise in representing, modeling, and creating visual information digitally, outcome *1a* tops out at **|**. Later, more advanced assignments will allow this outcome to expand to the maximum **+**.

Carlos Agudo

1a — Your `canvas` pictures show a solid ability to represent, model, and create visual information in terms of pixels and geometric primitives at a beginning, fundamental level. The proficiency will increase accordingly as long as you show continued advancement. (**|**)

4a — Your code is consistently correct and functional. You do have some hiccups in terms of JavaScript convention and style—I suggest that you send your code to JSLint to see where you can make your code more "JavaScript-y." To name a few: (1) put all of your variable declarations at the beginning of a function, under a single `var` statement and separated by commas; (2) favor `+= 1` or `-= 1` over `++` and `--`; (3) don't abuse the interchangeability of `"` and `'` when delimiting strings—pick one consistently, only choosing the other when absolutely necessary (e.g., like when the string has a delimiter character too).

Plus, what's up with your `for` loops in 28e? It's like you forgot how to use them there… (**|**)

4b — Your separation of concerns is mostly OK except for the inline CSS that you used for your `canvas` elements. I can see how you might have felt that separating this into its own `.css` file would have been overkill, but consider this: *you copy that same inline style for almost all of your pages*. If you had separated that out into its own `.css` file, then your `.html` files could all have referenced that. Net reduction in code, net increase in modifiability because you would only have to change that file if you wanted to adjust how your `canvas` elements look for this assignment. (**|**)

4c — There is much to improve in your code readability. Some specifics:

- Your spacing is too tight in places, particularly within expressions. Be more consistent about adding spaces after commas, around operators (especially binary ones including the assignment operator), and after semicolons. Add more blank lines to separate closely-related blocks of code.
- Your variable names—come on now, I'm *sure* Dr. Toal has shown you lots of examples for how to name your variables. They must be descriptive, and they should almost never have numeric suffixes. I added some inline comments for your more egregious ones from the assignment.
- And finally, your variables names. Did I mention your variables names? Remember about the variable names. Test yourself: after a few weeks, re-read your code (especially *28b* and *28e*). At each variable declaration, see if you can immediately state how that variable will be used.
- Oh yes, one more thing—*variable names*. (**/**)

4d — The final pictures lead me to believe that you made good use of available resources to create the images that you wanted to create. (**+**)

4e — You do phase your work, though not as I would have expected—instead of one picture at a time, you seem to be more batch-oriented. That can work, especially for the size and nature of this assignment, but your messages can definitely be improved "fixed *x*" or "cleaned up *y*" can be more specific (i.e., What exactly did you fix? What was cleaned up?). (**|**)

4f — Submitted on time, with some fixes after (although I can't tell you exactly what those fixes are based solely on the commit messages). (**+**)