CMSI 371-01

COMPUTER GRAPHICS

Spring 2012

Assignment 0306 Feedback

Lisa Rosenbaum

1a — Nothing committed yet, so will hold off. (**O**)

2b — Nothing committed yet, so will hold off. (**O**)

3a — Nothing committed yet, so will hold off. (**O**)

3b — Nothing committed yet, so will hold off. (**O**)

4a — Nothing committed yet, so will hold off. (**O**)

4b — Nothing committed yet, so will hold off. (**O**)

4c — Nothing committed yet, so will hold off. (**O**)

4d — Nothing committed yet, so will hold off. (**O**)

4e — Nothing committed yet, so will hold off. (**O**)

4e — Nothing committed. (**O**)

4f — Not submitted on time. (**–**)

Update 4/9/2012: I saw the work in progress for `primitives`, but without `nanoshop-filters` and answers to the questions, I decided it was still premature to update this. I did commit some comments to your primitives code inline, so do pull and look at what I said. Let me know when it's a good time for me to get back in here.

Updated feedback based on your commits up to 4/5/2014:

1a — Your variety of filters and primitives work shows a code-level understanding of bit-level graphic representation. Lack of answers to questions show a conceptual gap though. (**|**)

2b — Your work shows an understanding of the parts of a graphics system covered by this assignment. (**+**)

3a — You have shown that you can implement 2D-primitive-level graphics code somewhat. You didn't quite code the dashed-line functionality to spec (boolean vs. integer) and your fill-circle algorithm, although it does fill the circle, does so with extreme inefficiency at multiple levels. Finally, you did not do the fill-polygon exercise nor answered the conceptual questions, leaving a gap in your demonstration of this outcome. (**/**)

3b — Your filters show a good ability to manipulate colors (and images) at a low level. (**+**)

4a — Your code is functionally and syntactically correct, suffering primarily from the mismatched function signature for dashed lines, the sledgehammer approach in the filled circle, and the missing attempt at a gradient polygon fill. Filters are cleanly done though and `randomTint` in particular shows some nice out-of-the-box thinking, though not enough to make up for the issues with the primitives. (**/**)

4b — There is no evidence of additional comments from you in the code, but one of the comments directly affected by your modifications was not duly updated. Pay attention to these. (**/**)

4c — Your code is sufficiently readable and indented except for the fill-circle algorithm. That code block shows excessive repetition and insufficient spacing, exacerbating its already high complexity. (**|**)

4d — You show decent resource use but come up just short in the polygon-fill task. (**|**)

4e — One commit apiece for Nanoshop and primitives is not commensurate with the amount of work done. You need to be more granular. (**/**)