CS 410, Fall 2019, Last Third of the Semester Review
December 12, 2019

The following is a list of many of the main areas covered in the final third of the semester for CS410. The list covers material after the second Midterm. Keep in mind this list is not necessarily complete. However, it is offered in the hopes it will be helpful for students in general in when preparing for the final exam in particular.

1. You are comfortable with models of illumination that include both reflection and refraction. Refraction in particular is interesting because of the manner in which light passing through a solid is bent as defined by Snell's Law. Perhaps fully memorizing Snell's Law is unnecessary. However, being able to describe how light paths change upon transition between one material to another based upon indices of refraction must now come naturally.
2. There is a notable visual aid in thinking about refraction through a glass sphere based upon a photograph taken by Kelsey Rose Weber (Lecture 20 Slide 9). You will want to hold onto that image in your mind and be able to call upon it when thinking about refraction and explaining it to others.
3. You were not responsible for coding refraction through a polygonal solid in CS410. However, you are responsible for understanding some of the key elements of polygonal solids and refraction. Specifically, the idea of interpolating surface normals was discussed in lecture to 'smooth' the surface and you can at least explain the surface normal interpolation in broad terms. You also understand the impossibility of refracting through a bunny with holes (and you can generalize the problem from this description).
4. Ray tracing through a sheet of plate glass using refraction is possible and also boring. Knowing why it is boring is one side benefit of having worked through the geometry of refraction.
5. In CS410 you have not been asked to cast 'partial' shadows through semi-transparent objects. While you are therefore freed from the responsibility of committing to memory an algorithm for doing this type of light-through-objects calculation, you can easily explain at least one reason that doing this properly in the context of ray tracing is difficult.
6. Refraction through a sphere is a specialized and simplified case of general refraction through objects. One hallmark of this simplification is a specialized algorithm for finding the exit point for a refraction ray given the entry point. You can derive this specialized computation from first principles given the point where the ray is entering and the direction of travel through the sphere.
7. In Lecture 21 and the accompanying SageMath notebook a series of examples using a semi-transparent sphere with materials of differing indices of refraction are rendered. Through study of these images you can make relative judgements about the magnitudes of indices of refractions for spheres of differing materials based solely on how they bend light and hence behave in multi-object rendered scenes.
8. As is common in graphics, we go to great length to avoid ever calling on trigonometric functions in rendering code. Consequently, you now are familiar with at trigonometry free means of computing the direction of a refraction ray for light entering a material. The details in the algebra are complex enough that committing them to memory is less then worthwhile. However, given your understanding of the derivation, you are now able to recognize if a derivation being presented to you is correct.
9. Ray tracing with reflection and refraction implies that in a scene full of semi-transparent objects recursive calls may be modeled using a binary tree. You can use this mental-model to explain aspects of how your refraction enabled recursive ray tracer behaves.
10. Yet another material property is introduced when refraction is introduced; the opacity constant. Understanding how this constant, often abbreviated 'ko', means you can match up alternate scene renderings with different possible 'ko' values – provided the change in 'ko' is sufficient to be visually distinct.
11. Shadows are a shadowy concept in the presentation materials for CS410; often being nearly hidden from view. There is a reason for this shady treatment: shadows are easy for ray tracers. Just because not much gets said does not mean shadows are uninteresting or unimportant, and you have no trouble recognizing when they are present or modifying code to enable shadows should someone else have forgotten to do so.

12. Bicubic Parametric Curves are a fist taste of a rich family of curve and surface representations. Thanks to the Lectures at the end of this semester and the accompanying SageMath Notebooks you have a complete and thorough algebraic and geometry understanding of Hermite Curves, Bezier Curves, Bezier Surface patches, B-spline Curves and B-splines. Just for example, you would have no problem explaining to another person the geometric and algebraic setup for these shapes.
13. Often the CS410 instructor goes off on tangents, but seldom are tangents more on-topic than when introducing Hermite Curves. Indeed, with Hermite Curves, there is an entire physics metaphor involving time, direction of travel and speed, that can proof extremely helpful in developing an intuition for how these curves behave. Enjoy the fact that you can now work with this metaphor and even use it to think through some of the most common constraints one might want to impose when joining one curve to another.
14. Given the numerical setup of a curve or surface you could match it with a graphical illustration of the same curve provided the alternatives are reasonably distinct.
15. Given the Hermite, Bezier and B-spline blending functions, you could tell which is which. More important, you understand what is special about Bezier blending functions and why it matters. Likewise, you understand what is special about B-spline blending functions and why it matters.
16. The idea of de Casteljau Curves is extremely useful. While understanding the construction may in no way ever change how your write a single line of code, it will at some point prove invaluable for better understanding the nature of Bezier Curves. Because it is so important, be clear in your ability to explain these ideas to others.
17. In Lecture 24 a distinction was made between the dimensionality of a geometric object such as a curve or surface and the embedding dimension. Hang onto this distinction and be ready to use it when explaining curves and surfaces to others.
18. The notion of degrees of freedom plays an important role in parametric curves and surfaces. So, for example, you can now discuss the degrees of freedom intrinsic to a 3D Bezier curve.
19. Bezier curves and surfaces must pass through some control points and not others. You now understand which are which.
20. B-spline segments, while never required to pass through control points, are designed to work together when using more a larger sequence of control points. You are now able to work out how successive segments are blended together to form longer spline curves.
21. When drawing shapes in many common drawing packages, PowerPoint comes to mind, many of you have seen ‘wings’ attached to points in which growing, shrinking, and rotating these wings changes the curve. While before this was all magic, now you see this instantly as an instantiation of Bezier Curves.
22. While NURB surfaces received scant real treatment in CS410 this semester, there is one truly important property of NURBs that arises in the context of traditional pin-hole camera rendering. Make sure you at least are aware of this property and why it is of such practical importance.