CSCD18 Computer Graphics, Fall 2007
Assignment 4

Due online by 11:59pm on Monday, December 3 [50 marks]
Demo day, December 3 (in class)

For the final project, you will implement a more advanced ray-tracer. You are expected to work in teams of two students, but it is acceptable to work alone. You may divide up the work on the assignment in any way you see fit. However you must acknowledge who did what in the write-up. Both members will receive the same mark, and of course both team members should understand the entire program, if for no other reason than your preparation for the final exam.

The programming assignment is somewhat open-ended, so be careful to manage your time, implement one element at a time to simplify debugging, and focus on the most important elements and those you understand clearly. If your solution is well organized before you begin to code and debug it, you should be able to complete it within a reasonable number of hours. There is no starter code for this assignment, but you are encouraged to reuse code from the previous assignment. If your programs for previous assignment did not work, please talk to the professor immediately.

Extend the basic ray tracer that you wrote for Assignment 3 to include the following features:

- Recursive ray tracing to show secondary reflection with specularities so you can see one object reflected in another
- Shadows
- Any **two** of the following, at least **one** from each group:
  - Geometry
    - Additional 3D object type(s) created with quadratic surfaces (*e.g.* cone or cylinder)
    - Constructive Solid Geometry (CSG) for surfaces that are created by subtracting or adding two quadratic surfaces (*e.g.* subtraction of one cylinder from another)
    - Handling of arbitrary surface mesh geometry
  - Lighting and surface properties
    - Refraction (*e.g.* glass spheres)
    - Texture-mapping. An interesting procedural texture is also acceptable.
    - Anti-aliasing
    - Extended light sources to produce soft shadows
    - Apply (non trivial) BRDF to one or more of the surface geometries

You are encouraged to be creative in creating an interesting and/or attractive scene. You are allowed to use geometry that is obtained from the web (*i.e.* you can download surface meshes), but the code to render that geometry must be written by yourself.

The program you hand in should produce the final result (which may be slow to compute). In addition to your program, hand in the following renderings of your scene:

- scene signature (filename: `sceneSig.bmp`)
- diffuse rendering (with no recursive ray bouncing) (filename: `diffuse.bmp`)
- diffuse rendering with shadows. (filename: `diffuseShadow.bmp`)

- recursive ray tracer with the recursive depth set to 1 and 2. (filename: `recurs1.bmp` and `recurs2.bmp`)
- your final result (filename: `final.bmp`)

Tip: Keep the recursion depth low (2 or 3) to speed up debugging. We recommend that you implement this project in stages, rather than attempting everything at once. *E.g.* make sure object geometry and intersections are correct. Then add other effects (like local shading, shadows, secondary effects, and texture mapping) one at a time.

Marking:

The work you do for this assignment should be done solely by your team members. Any geometry downloaded from the web should be properly acknowledged. You may not use code from the web or other sources without advance permission.

All submission for this assignment is electronic. In your submission, please include a README file (as TXT or PDF) that provides the following:

1. A well-written, high level description of what your program does, and which effects you implemented.
2. A brief list of where to look in the images/movies you submit to see the requirements of the assignment, *e.g.* where to see refraction, shadows, *etc.* Be sure to choose images that demonstrate the requirements.
3. If you implemented anything notable beyond the assignment, explain what you did and where to see it in the image/movies.
4. A listing of all of the files submitted.
5. For images and movies, a brief description of what each one shows.

This should not be much more then what you present and talk about during demo day. In fact, we will accept extended versions of your demos or slides (that contain all of the information requested above) in lieu of the more formal write-ups.

Bonus marks. Up to 10 additional marks will be awarded for exceptional solutions. This will include any special technical aspects that are well above and beyond the technical requirements specified above. Bonus marks will also be given for exceptionally creative or artistic work.