

Assignment 1: Simple Ray Tracing

Implement simple ray tracing algorithm that computes rays for a parallel and perspective projection for a given fixed viewpoint and viewing plane of a resolution at least 640x480. Viewing plane should be defined by equation $z = 0$ and its extents are $[-2, 2]$ in the x direction and $[-1.5, 1.5]$ in the y direction with the viewing direction in the positive z direction.

The viewpoint should be at the position $(0, 0, 1)$.

There are some objects (plane, sphere, ellipsoid, cone, cylinder) defined in the scene. Perform a simple intersection test for all pixels iterating over all objects.

For each pixel define the closest point of intersection and calculate the surface normal at this location.

Then perform the shading with the two lights:

- light1 positioned at the location $(0, 5, 0)$ with light color $(1, 1, 1)$;
- light2 positioned at the location $(5, 5, 0)$ with light color $(1, 1, 1)$.

Calculate a simple Lambertian + Phong Shading Model at each surface point with $k_a = 0.2$, $k_d = 0.6$, $k_s = 0.2$, Phong exponent = 32

Keep in mind that each shadow ray (ray from surface point to the light source point) you need to calculate whether it intersects another object or not.

If it does, the object is in shadow and only the ambient component will be used in the illumination calculation.

For each calculated pixel value cast/convert the floating point values to the fixed point range of $[0..255]$ for each R,G,B channel.

To do all the tasks above, find all #TODO tags in the source code and complete them.