# Algorithm 5: Ray

Comp175: Introduction to Computer Graphics – Spring 2016

Algorithm due: **Monday April 11th** at 11:59pm

Your Names: __Phoebe Yang, Alex Cohen, Holt Spalding

Your CS Logins: __yyang08, acohen05, hspald01_

## Instructions

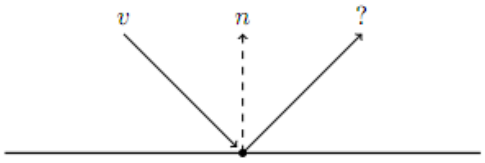
Complete this assignment only with your teammate. When a numerical answer is required, provide a reduced fraction (i.e. 1/3) or at least three decimal places (i.e. 0.333). Show all work; write your answers on this sheet. This algorithm handout is worth 3% of your final grade for the class.

**[2 points]** The high-level view of our ray tracer is exactly the same as for intersect, except for a few additions. Below is the high-level pseudocode for `Intersect`. What needs to be changed/added to make this a full-fledged ray-tracer? Just specify what changes need to be made no pseudocode please.

**for** *point* $\in$ *Canvas* **do**
  Cast a ray to find the nearest object
  **if** ray intersects an object **then**
    **for** each light **do**
      Cast a ray to the light and evaluate the lighting equation
      *Canvas*[*pt*] = *Canvas*[*pt*]+color with only diffuse/ambient components
    **end for**
  **else**
    *Canvas*[*pt*] = background color
  **end if**
**end for**

We must add the reflected and transmitted (for refraction) factors to our lighting equation:
reflected factor: $k_s * O_s * I_r$
transmitted factor: $k_t * O_t * I_t$

**[2 points]** Given a vector $\vec{v}$ and a surface normal $\vec{n}$, find the equation for the vector $\vec{r}$ which is the reflection of $\vec{v}$ about $\vec{n}$ (i.e. in the equal and opposite direction). Write your equation in terms of vector operations. How do you compute the color contributed by the reflected ray? Give a brief description.




r = v - 2*dot(v,n)*n
r's color contribution is $k_s * O_s * I_r$ where $k_s$ is the specular constant, $O_s$ is the specular channel, and $I_r$ is calculated recursively. Recursion terminates when either the maximum recursive level is reached (eg. I=3), or the global contribution falls below a given threshold.

**[1 point]** Is ray tracing a local or global illumination algorithm? Why?

Global; it uses direct illumination and takes into account shadows and indirect illumination (reflection/refraction).

**[1 point]** For a particular ray that intersects with an object, when do you not consider con-

tribution from a given light source? How do you computationally determine when this scenario occurs?

When there's another object blocking the given objects path to the light source (ie. another object is casting a shadow); You determine this by casting a ray from the object to a light source and only count the light's full contribution if the ray does not intersect with any object on its way to the light source (note that this method causes hard shadowing). When dealing with recursive raytracers which consider indirect contributions of a light source, you may choose not consider the contribution from a given light source, ie terminate the recursive step of the calculation, when: 1) a light sources contribution falls below a given threshold, or 2) max reursive depth is reached. We can also skip cacluating reflection rays for non-reflective materials, and refraction for opaque materials.

[**2 points**] Recall that we can think of texture mapping in two steps. First, mapping from the object to the unit square, and second, mapping from the unit square to the texture map. Let $u$ and $v$ be the $x$ and $y$ values in the unit square that a particular point on an object gets mapped to in the first step. Note that $a$ and $b$ are calculated differently depending on the object. From here, how do you find the coordinates $(s, t)$ to look up in a texture map in terms of $u, v, i, j, w$ and $h$, where $i$ and $j$ are the number of repetitions in the $x$ and $y$ directions, respectively, $w$ is the texture width, and $h$ is the texture height?

(s,t) = (i/u*w,j/v*h)

[**1 point**] How do you use the color from the texture map and the `blend` value in the lighting equation?

(pixel color of texture map)*blend + (pixel color of object)*(1-blend)

[**1 point**] What is the Phong lighting model used for? What is the purpose of its exponent?

For calculating specular light; Exponent f accounts for the surface property for specular highlight - great the f, the more specular the surface is.

## How to Submit

Hand in a PDF version of your solutions using the following command:

```
provide comp175 a5-alg
```