Com S 336
Fall 2015
Homework 4

1. The file Lighting4.js is similar to Lighting3.js, except that a plane is also drawn below the rotating model to look like a "floor". In addition, there is an instance of CS336Object that is used for the light position, so you can move it around with the key controls. Try it out. There is some console output from the key handler showing the light's current position. It also prints the "forward direction" of the CS336Object (negative z-axis) but this vector currently has no effect on the scene, since the only the light's position is being used.

There is also a file Lighting4.html. You will need to make some minimal modifications to both files for this problem. These files are in the **`lighting`** directory, and you can find a working CS336Object.js there too.

Your task is to make the point light into a spotlight. One simple way of defining a spotlight is with two parameters (that you will need to define as uniform variables) in addition to the light position:

- A direction, *D*, which is a unit vector representing the direction in which the center of the light cone is pointing
- An exponent, *s*, similar to the "shininess" exponent in the ADS reflection model, that determines how the light intensity decreases as points move further from the center of the light cone

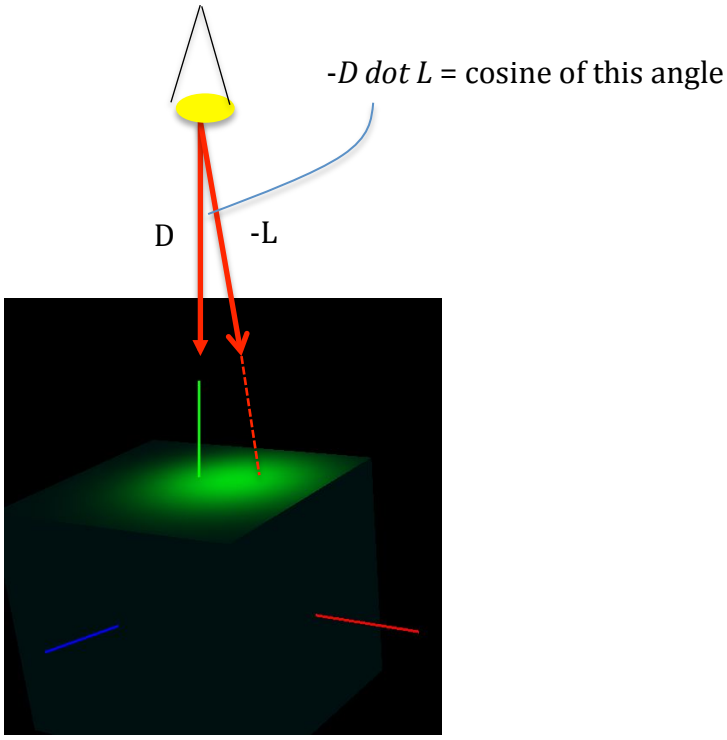
If *L* is the unit vector from a fragment's position toward the spot position, then *–D dot L* is the cosine of the angle between that vector and the direction of the spotlight. See illustration on next page. The quantity

$$(max\ (-D\ dot\ L)\ ,\ 0))^s$$

can then be used to modulate the light's contribution to the fragment's color. "Modulate" just means that you have a number between 0 and 1 that you are using to multiply some existing value. **In particular, you do all the ambient + diffuse + specular calculations normally, and then apply this factor only at the end so that fragments outside the light's cone are darker.** The exponent *s* determines how tightly focused the spot cone is. (Also remember that we calculate L in eye coordinates, so for the dot product to make sense, the vector D had better be transformed into eye coordinates too!)

The existing keyboard controls can be used to "aim" the light. Add two more keyboard controls: use 'c' to increase the exponent and 'C' to decrease it, which should make the cone more focused or less focused.

In the picture below, the light position is directly overhead at (0, 2, 0) and the spot direction is [0, -1, 0] (pointing straight down). (The exponent is 40.)

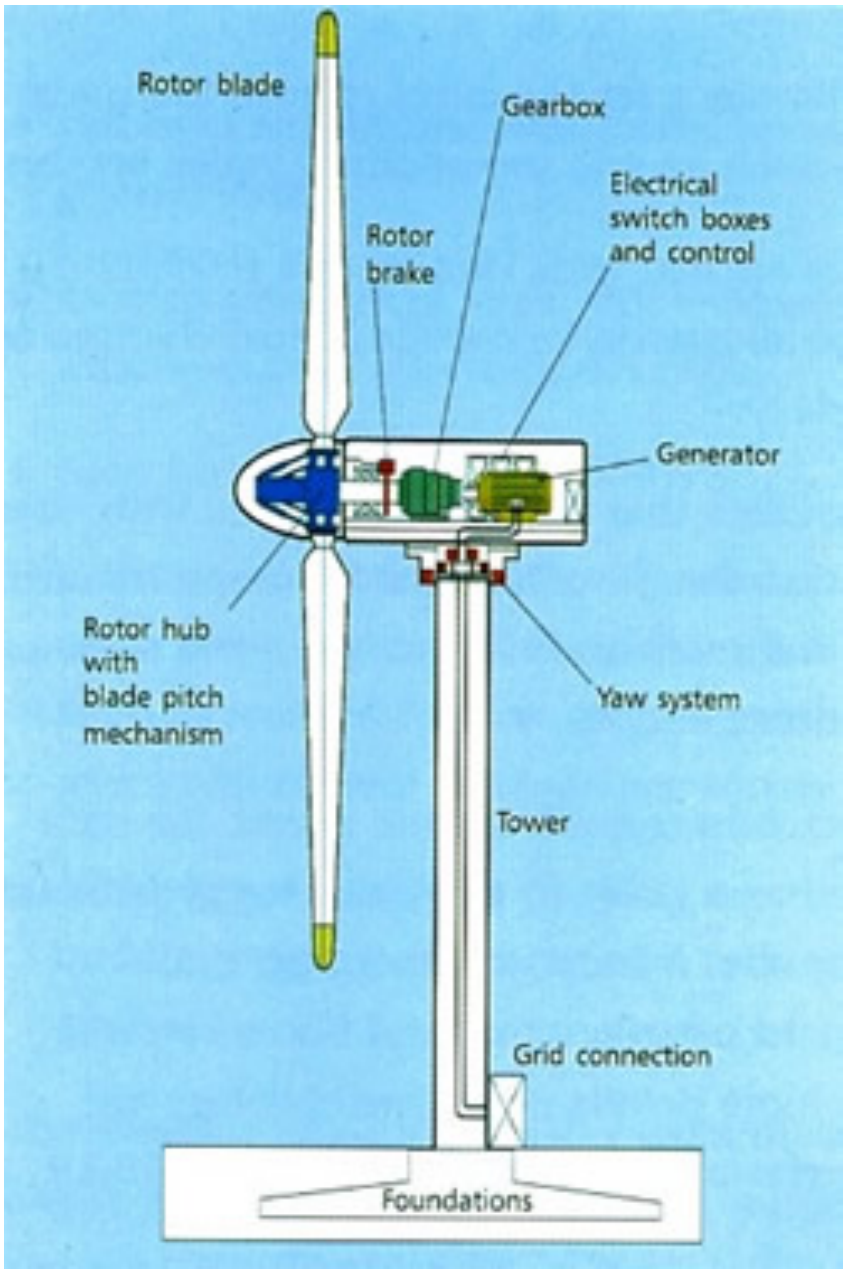


2. Create a hierarchical scene using the techniques of HierarchyWithTree3.js. Your basic task is to create a clunky model of a wind turbine using five scaled cubes. Parts should include

- a vertical shaft,
- a generator housing at the top of the shaft,
- a rotor hub, and
- two rotor blades.

The housing should turn on the vertical shaft (the "yaw" control to make it face the wind), the rotor should rotate, and the blades themselves should rotate relative to the rotor to adjust the "pitch" of the blades (angle relative to the rotor). Animate the rotor to rotate continuously, and add some keyboard controls for direction and blade pitch. (If you are feeling ambitious, make the rotation speed dependent on the blade pitch.)

On the next page is a basic diagram.


Rotor blade
Gearbox
Electrical
switch boxes
and control
Rotor
brake
Generator
Rotor hub
with
blade pitch
mechanism
Yaw system
Tower
Grid connection
Foundations