Render Modes
- glRenderMode(mode)
  - GL_RENDER: normal color buffer
  - default
  - GL_SELECT: selection mode for picking
  - (GL_FEEDBACK: report objects drawn)

Project 2 Hints
- don't forget to keep viewing and projections in their respective stacks
- try drawing scene graphs to help you figure out how to place multiple cameras
  - especially geosynchronous: camera as child of object in world in the scene graph
  - geometric representation of camera vs. what is shown through its window
- disk for Saturn rings: try scaling sphere by 0
- OK to reset camera position between absolute/relative navigation modes
- OK to have camera jump to different orientation when new planet picked in geosync mode

News
- Homework 2 out
  - due Fri Feb 12 5pm
- Project 2 out
  - due Tue Mar 2 5pm
- moved due date after break after pleas of pre-break overload with too many assignments due
- start early, do "not" leave until late in break!
- reminder
  - extra handouts in lab

Review/Clarify: Trackball Rotation
- user drags between two points on image plane
  - mouse down at \( (x, y) \), mouse up at \( (a, b) \)
- find corresponding points on virtual ball
  - \( p_1 = (x, y, z) \), \( p_2 = (a, b, 0) \)
- compute rotation angle and axis for ball
  - axis of rotation is plane normal: cross product \( p_1 \times p_2 \)
  - amount of rotation \( t \) from angle between lines
  - \( t \cdot p_2 = [t \cdot [p_1, [p_2, C]]] \)

Review/More: Relative Motion
- how to move relative to current camera?
  - what you see in the window
  - computation in coordinate system used to draw previous frame is simple:
    - incremental change \( i \) to current \( C \)
    - each time we just want to premultiply by new matrix
      - \( p = C \cdot p \)
      - but we know that OpenGL only supports postmultiply by new matrix
  - use OpenGL matrix stack as calculator/storage!
  - dump out model/view matrix from previous frame with glGetDoublev()
  - \( C = \) current camera coordinate matrix
  - wipe the matrix stack with glLoadIdentity()
  - apply incremental update matrix \( i \)
  - apply current camera coord matrix \( C \)

Review: Select/Hit Picking
- use small region around cursor for viewport
- assign per-object integer keys (names)
- redraw in special mode
- store hit list of objects in region
- examine hit list
  - OpenGL support

Viewport
- small rectangle around cursor
  - change coord sys so fills viewport
  - why rectangle instead of point?
    - people aren't great at positioning mouse
    - Fitts' Law: time to acquire a target is function of the distance to and size of the target
    - allow several pixels of slop

Name Stack
- again, "names" are just integers
  - glInitNames()
- flat list
  - glLoadName(name)
  - or hierarchy supported by stack
  - glPushName(name), glPopName
  - can have multiple names per object

Hierarchical Names Example
```c
for(i = 0; i < 2; ++i)
  {
    glPushMatrix();
    key = i == 0 ? 1 : 2; ++i;
    glPushMatrix();
    glPushMatrix();
    glLoadIdentity();
    glCallList(snowManBodyDL);
    glCallList(snowManHeadDL);
    glCallList(snowManBodyDL);
    glCallList(snowManHeadDL);
    glPopName();
    glPopMatrix();
  } // end for loop.
```

Hit List
- glSelectBuffer(buffersize, "buffer")
  - where to store hit list data
- on hit, copy entire contents of name stack to output buffer.
- hit record
  - number of names on stack
  - minimum and maximum depth of object vertices
  - depth lies in the NDC z range \([0,1]\)
- format: multiplied by \(2^{32}-1\) then rounded to nearest int
17 Integrated vs. Separate Pick Function
- integrate: use same function to draw and pick
  - simpler to code
  - name stack commands ignored in render mode
- separate: customize functions for each
  - potentially more efficient
  - can avoid drawing unpickable objects

18 Select/Hit
- advantages
  - faster
  - OpenGL support means hardware acceleration
  - avoid shading overhead
  - flexible precision
  - size of region controllable
  - flexible architecture
  - custom code possible, e.g. guaranteed frame rate
- disadvantages
  - more complex

19 Hybrid Picking
- select/hit approach: fast, coarse
  - object-level granularity
- manual ray intersection: slow, precise
  - exact intersection point
- hybrid: both speed and precision
  - use select/hit to find object
  - then intersect ray with that object

20 High-Precision Picking with OpenGL
- gluUnproject
  - transform window coordinates to object coordinates
given current projection and modelview matrices
- use to create ray from cursor location
call gluUnProject twice with same (x,y) mouse location
  - z = near: (x,y,0)
  - z = far: (x,y,1)
- subtract near result from far result to get direction vector for ray
  - use this ray for line/polygon intersection

21 Additive vs. Subtractive Colors
- additive: light
  - monitors, LCDs
  - RGB model
- subtractive: pigment
  - printers
  - CMY model
  - dyes absorb light

22 Electromagnetic Spectrum
- common light sources differ in kind of spectrum they emit:
  - continuous spectrum
    - energy is emitted at all wavelengths
    - blackbody radiation
    - tungsten light bulbs
    - certain fluorescent lights
    - sunlight
    - electrical arcs
  - line spectrum
    - energy is emitted at certain discrete frequencies

23 Vision/Color
- elements of color:
  - physics
  - illumination
  - electromagnetic spectra
  - reflection
  - material properties
  - surface geometry and microgeometry
    - polished versus matte versus brushed
  - perception
    - physiology and neurophysiology
    - perceptual psychology

24 Additive vs. Subtractive Colors
- additive: light
  - monitors, LCDs
    - RGB model
- subtractive: pigment
  - printers
  - CMY model
  - dyes absorb light

25 Component Color
- component-wise multiplication of colors
  - \((a_0, a_1, a_2) \times (b_0, b_1, b_2) = (a_0b_0, a_1b_1, a_2b_2)\)
- Light + object = color
  - why does this work?
    - must dive into light, human vision, color spaces

26 Blackbody Radiation
- black body
  - dark material, so that reflection can be neglected
  - spectrum of emitted light changes with temperature
    - this is the origin of the term “color temperature”
    - e.g. when setting a white point for your monitor
  - cold: mostly infrared
  - hot: reddish
  - very hot: bluish
  - demo:

27 Electromagnetic Spectrum
- 700 nm - 400 nm
- AM radio/microwave ultraviolet x-rays
- FM radio, TV
- infrared

28 Basics Of Color
- elements of color:
  - physics
    - illumination
    - electromagnetic spectra
    - reflection
    - material properties
    - surface geometry and microgeometry
      - polished versus matte versus brushed
    - perception
      - physiology and neurophysiology
      - perceptual psychology

29 Light Sources
- common light sources differ in kind of spectrum they emit:
  - continuous spectrum
    - energy is emitted at all wavelengths
    - blackbody radiation
    - tungsten light bulbs
    - certain fluorescent lights
    - sunlight
    - electrical arcs
  - line spectrum
    - energy is emitted at certain discrete frequencies
  - bulb spectrum
    - tungsten filament
    - energy is emitted at specific wavelengths
  - line spectrum
    - energy is emitted at specific wavelengths

30 Additive vs. Subtractive Colors
- additive: light
  - monitors, LCDs
  - RGB model
- subtractive: pigment
  - printers
  - CMY model
  - dyes absorb light

31 Component Color
- component-wise multiplication of colors
  - \((a_0, a_1, a_2) \times (b_0, b_1, b_2) = (a_0b_0, a_1b_1, a_2b_2)\)
- Light + object = color
  - why does this work?
    - must dive into light, human vision, color spaces

32 Electromagnetic Spectrum
- 700 nm - 400 nm
- AM radio/microwave ultraviolet x-rays
- FM radio, TV
- infrared
**White Light**
- sun or light bulbs emit all frequencies within visible range to produce what we perceive as "white light"

**Sunlight Spectrum**
- spectral distribution: power vs. wavelength

**Continuous Spectrum**
- sunlight
- various "daylight" lamps

**Line Spectrum**
- ionized gases
- lasers
- some fluorescent lamps

**White Light and Color**
- when white light is incident upon an object, some frequencies are reflected and some are absorbed by the object
- combination of frequencies present in the reflected light that determines what we perceive as the color of the object

**Tristimulus Theory of Color Vision**
- Although light sources can have extremely complex spectra, it was empirically determined that colors could be described by only 3 primaries
- Colors that look the same but have different spectra are called metamers

**Hue**
- hue (or simply, "color") is dominant wavelength/frequency
- integration of energy for all visible wavelengths is proportional to intensity of color

**Saturation or Purity of Light**
- how washed out or how pure the color of the light appears
- contribution of dominant light vs. other frequencies producing white light
- saturation: how far is color from grey
  - pink is less saturated than red
  - sky blue is less saturated than royal blue

**Intensity vs. Brightness**
- intensity: physical term
- measured radiant energy emitted per unit of time, per unit solid angle, and per unit projected area of the source (related to the luminance of the source)
- lightness/brightness: perceived intensity of light
  - nonlinear

**Physiology of Vision**
- Center of retina is densely packed region called the fovea.
- Cones much denser here than the periphery
- Center of retina is densely packed region called the fovea.

**Trichromacy**
- three types of cones
  - L or R, most sensitive to red light (610 nm)
  - M or G, most sensitive to green light (560 nm)
  - S or B, most sensitive to blue light (430 nm)

**Color Spaces**
- three types of cones suggest color is a 3D quantity. how to define 3D color space?
  - idea: perceptually based measurement
    - shine given wavelength (λ) on a screen
    - user must control three pure lights producing three other wavelengths
      - used R=700nm, G=546nm, and B=436nm
    - adjust intensity of RGB until colors are identical
      - this works because of metamers!
    - experiments performed in 1930s
Negative Lobes

- sometimes need to point red light to shine on target in order to match colors
  - equivalent mathematically to "removing red"
  - but physically impossible to remove red from CRT phosphors
- can’t generate all other wavelengths with any set of three positive monochromatic lights!
- solution: convert to new synthetic coordinate system to make the job easy

CIE Color Space

- CIE defined 3 ‘imaginary’ lights X, Y, Z
  - any wavelength λ, can be matched perceptually by positive combinations

Note that:
X = R
Y = G
Z = B

Measured vs. CIE Color Spaces

- X, Y, Z form 3D shape
  - project X, Y, Z on X+Y+Z=1 plane for 2D color space
  - chromaticity diagram
  - separate color from brightness
  - x = X / (X+Y+Z)
  - y = Y / (X+Y+Z)

CIE “Horseshoe” Diagram Facts

- all visible colors lie inside the horseshoe
  - result from color matching experiments
  - spectral (monochromatic) colors lie around the border
  - straight line between blue and red contains purple tones
- colors combine linearly (i.e. along lines), since the xy-plane is a plane from a linear space

Color Interpolation, Dominant & Opponent Wavelength

- gamut is polygon, device primaries at corners
  - defines reproducible color range
  - X, Y, and Z are hypothetical light sources, no device can produce entire gamut

HSV Color Space

- more intuitive color space for people
  - H = Hue
  - dominant wavelength, “color”
  - S = Saturation
  - how far from grey/white
  - V = Value
  - how far from black/white
  - also brightness, B, intensity, I, lightness L
- also: brightness B, intensity I, lightness L

RGB Color Space (Color Cube)

- define colors with (r, g, b)
  - amounts of red, green, and blue
    - used by OpenGL
    - hardware-centric
  - RGB color cube sits within CIE color space
  - subset of perceivable colors
  - scale, rotate, shear cube

Display Gamuts

- all positive, unit area
  - Y is luminance, no hue
  - X, Z no luminance

Gamut Mapping

- how to handle colors outside gamut?
  - one way: construct ray to white point, find closest displayable point within gamut

CIE and Chromaticity Diagram

- can choose a point C for a white point
  - corresponds to an illuminant
  - usually on curve swept out by black body radiation spectra for different temperatures
  - two colors are complementary relative to C when are
    - located on opposite sides of line segment through C
    - C is an affine combination of the two colors
  - find dominant wavelength of a color
    - extend line from C through color to edge of diagram
    - some colors (i.e. purples) do not have a dominant wavelength, but their complementary color does

Blackbody Curve

- illumination:
  - candle 2500K
  - A: Light bulb
  - B: Sunrise/sunset
  - C: Dusk/dawn
  - D: Daylight 6500K
  - E: Overcast day 7000K
  - Light from H: >20,000K

Projector Gamuts

- all positive, unit area
  - Y is luminance, no hue
  - X, Z no luminance

HSV/HSI and RGB

- HSV/HSI conversion from RGB not expressible in matrix
  - H=Hue same in both
  - V=Value is max, Intensity is average

\[
H = \cos^{-1}\left(\frac{1}{2}(R - G) + (R - B)\right) \quad \text{if } (B > G), \quad H = 360 - H
\]

HSI:

\[
S = 1 - \frac{\min(R, G, B)}{I} \quad \text{if } (R + G + B) \neq 0

V = \max(R, G, B)
\]