CS420 Assignment 2 Extra Hints
How to draw a rectangle cross-section

- For each point \( p(u) \) on the curve
  - Compute local coord: \( T, N, B \)
  - Find 4 points \( v_0(u), v_1(u), v_2(u) \) and \( v_3(u) \) on the N-B plane as the vertices for a rectangle cross-section

- Draw cross-section between \( u_0 \) and \( u_1 \) by connecting \( v_0(u_0), v_1(u_0), v_2(u_0), v_3(u_0) \) and \( v_0(u_1), v_1(u_1), v_2(u_1), v_3(u_1) \) with triangles
Lighting

• Init lighting:
  - `glLightfv(GL_LIGHT0, GL_DIFFUSE, ...)`
  - Same for ambient and specular color

• In the display loop:
  - `glMaterialfv(GL_FRONT, GL_DIFFUSE, ...)`
  - Same for specular, ambient and shinness
  - `glLightfv(GL_LIGHT0, GL_POSITION, ...)`
  - `glEnable(GL_LIGHTING)`
- Draw objects
- If following objects don't need lighting:
  - `glDisable(GL_LIGHTING);`
  - `glLightfv(GL_LIGHT0, GL_POSITION,...)`
- `MODELVIEW` matrix will affect the position of light sources
• Init texture:
  - `glGenTextures(1, &textureName);`
  - Load image data
  - `glBindTextures(GL_TEXTURE_2D, textureName);` //tells openGL subsequent code all works on the texture with the name `textureName` until another `glBindTextures` with a different texture name is called
- `glTexParameteri`:
  - `GL_TEXTURE_WRAP_S/T`
  - `GL_TEXTURE_MIN/MAG_FILTER`
- `glTexImage2D(GL_TEXTURE_2D, GL_RGBA, width, height, GL_RGB, GL_UNSIGNED_BYTE, pointer)`;
- Or: `GluBuild2DMipmaps to use mipmaps`
• In the display loop:
  
  - `GLTexEnvf(...,GL_REPLACE/BLEND/MODULATE);` //tells openGL the method to combine texture and lighting in subsequent code until another `GLTexEnvf` changes the method
  
  - `glEnable(GL_TEXTURE_2D);`
  
  - `glBindTextures(GL_TEXTURE_2D,texture Name);`
- Use `glTexCoord2f(0.0,0.0)` to specify texture coordinates for vertices when drawing
- If the following objects don't need texture:
  - `glDisable(GL_TEXTURE_2D);`
Recursive Subdivision

- $\text{subdivide}(u_0, u_1, \text{maxlinelength})$
  - $\text{umid} = \frac{u_0 + u_1}{2}$;
  - Compute curve point position $p_0$ and $p_1$ corresponding to $u_0$ and $u_1$
  - If distance between $p_0$ and $p_1$ is larger than $\text{maxlinelength}$
    - $\text{subdivide}(u_0, \text{umid}, \text{maxlinelength})$
    - $\text{subdivide}(\text{umid}, u_1, \text{maxlinelength})$
  - Else $\text{drawline}(u_0, u_1)$
• Call this subdivide function inside the drawing function or glCallList:
  – maxlinelength = 0.001;
  – subdivide(0,1,maxlinelength);
Rail Artifact

- Rail curvature is too large
- Could be avoided by changing orientation
- ...
Sky Dome

• Using latitude $\phi$ and longitude $\theta$ angles

Dome Texture

- Two kinds of texture image
Camera Speed

- $h_{\text{max}}$: the maximum potential energy

$$u_{\text{new}} = u_{\text{current}} + (\Delta t) \frac{\sqrt{2g(h_{\text{max}} - h)}}{||dp/du||}$$

- $E_v + E_p = E_{p_{\text{max}}}$
- $||v|| = ||dp/dt||$
Thanks!