LECTURE 9
Announcements
Platformer3 Feedback

1 = 😊
2 = 🙆

Warmup, FPS: 56.285
Next week...

- Does not exist (as far as our class is concerned)
- This means Platformer4 is due the Tuesday after Spring Break 4/4
- All retries are also not due until that day
No more design checks!

- Platformer4 doesn’t have any design checks
- Enjoy your extra 15 minutes a week!
Final Projects

- Ideas due today
- Will be sent out to everyone in the class
- Email each other and decide on groups over break
- Groups and final proposal due on 4/4 (same day as Platformer4)
Mid-semester Feedback!

• We’d like to know how we’re doing so far
  – Fill out the form here

• We’d also like to know what additional topics you want to learn about
  – Fill out form here
LECTURE 9
User Interfaces
(Common Engine)
User Interfaces (Common Engine)

USER INTERFACES
Many shades of UI

- Heads-up display (HUD)
  - Persistent 2D elements drawn over the game
    - Ex. your health & resources, minimap, available actions/cooldowns

- Separate screens/overlays
  - Inventory, menus, shops, dialog

- Orthographic UI
  - 2D elements drawn in the 3D scene
    - Ex. entity health bars, tips, popup dialog, 2D special effects
    - Usually doesn’t scale with camera distance
Heads-Up Display

Player health & abilities (most important)

Player stats & inventory

Game stats (KDA)

Minimap

HUD

(League of Legends)
HUD Clarity

• The HUD takes up minimal screen space
• Each different type of information is placed in a distinct and isolated area of the screen
HUD Clarity

- Deciding where to put information (and how to group it) is very important
  - In League, there can be many passive effects on your character at once

‘Other’ effects go here

‘Battle’ effects go here
HUD Clarity

- But League is a battle arena (MOBA) game, so almost every effect is battle-related
  - The developers had to split them somewhat arbitrarily, which caused a lot of confusion
Orthographic UI

Tooltips

Health bars & Damage

(World of Warcraft)
Orthographic UI

"MEDIC!"
(Team Fortress 2)

Dialog
(Runescape)
Converting to screen space

```cpp
vec3 convertToScreenSpace(vec3 pos) {
  // get these from your camera
  mat4x4 view, projection;
  // this is necessary for the matrix multiplication
  vec4 fourVec = vec4(pos.x, pos.y, pos.z, 1.f);
  // project the position into screen space
  fourVec = projection * view * fourVec;
  return vec3(fourVec.x / fourVec.w, fourVec.y / fourVec.w, fourVec.z / fourVec.w);
}
```
Using the projected position

- Use the x- and y-components of the vector calculated in the previous slide to position draw calls on the screen.
- If the z-component is less than 0, it means that the object is behind the camera. In other words, don’t draw it!
1. Create a VAO/VBO that holds data for a single quad that takes up [-1, 1] in the xy-plane (leave z = 0)
   - This is essentially a quad that takes up the whole screen
   - Make sure it includes texture coordinates!
2. Use the projection math from earlier on the position vector to calculate the screen-space position

3. Check if the z-component of the screen-space position is greater than zero before drawing

4. Create a model matrix:
   - Translate by \((sx, sy, 0)\)
     - \(sx = \) screen position x-component
   - Scale by a user-specified amount
     - Likely a small amount of the screen space, like \((5\%, 5\%)\)
     - This determines how much screen space the UI element will take up
5. Send the model matrix to the shader
6. Set the view and projection matrices to be the identity matrix (default mat4x4 constructor)
7. Bind a texture and draw it to the screen!
Integrating this?

Add it to the basic Application!

class Application {
public:
    void onTick(long nanos);
    void onDraw(Graphics *g);
    void onUI(Graphics *g); // same thing in Screen, etc...
    void onKeyPressed(QKeyEvent *event);
    void onMouseDragged(QMouseEvent *event);
}
Your Orthographic UI

• You are required to implement some use of orthographic UI in platformer 4
  — Try to think of something unique and interesting to your game concept!
    • (Health bars are still acceptable)

• You could also make even more awesome UI...
3D-Integrated UI

Instructions
(Splinter Cell)

Minimap
(Far Cry 2)
3D-Integrated UI

All user interfaces as in-game holograms

(Dead Space)
User Interfaces

QUESTIONS?
PLATFORMER3 PLAYTESTING!

Fill out the Mid-Semester feedback form too!