Fine-Tuning Game Controls
Upcoming deadlines

- **Friday, February 8**
  - Team status report due, via team status report tool

- **Friday, February 22**
  - Sprint 2 ends
  - 14 days until the end of the Sprint (including President’s Day holiday)
Team meetings

- Will be meeting Siegebreakers (3:30pm) and Lens (4:30pm) today
Lab Cleanup Schedule

- This week: We’re Screwed!
- Next week: 10 Days Left to Live

Team duties:
- Ensure overflowing trash cans are emptied to bin outside in 3rd floor courtyard (anytime during week)
- By 5pm Monday and 5pm Friday (unless things get out of control, then more often):
  - Pick up food containers, bottles, etc.
  - Pick up stray craft materials, pens, etc and return to drawers
  - Clean off tables in conference rooms and big circular table
  - Report any major soda/food spills to me, so we can call cleanup crews
  - Put controllers/game boxes/etc. away (tidy up game area)
  - Report any cleaning materials needed
Next week’s meetings: Art Direction Review

- Starting with next week’s meetings, want to review the art direction for each game
- Please come to meeting with examples of concept art, artwork for items in the game, etc.
- General goal is that I want to get a good feel for what the final look of the game will be, and provide feedback
- Artists are welcome and encouraged to attend, but if they can’t make it, that’s fine too
Getting movement control right

- Several common control problems with games
  - Floaty: after receiving player input, the player avatar is non-responsive for a short period of time
    - Avatar is “floating” through space
  - Twitchy: player avatar is overly responsive to player input
    - At its worst, avatar is vibrating, reacting even to noise in reading values of the physical controls
  - Unresponsive: player avatar responds slowly, or unreliably to player input
Controller inputs are mapped to in-game parameters
- Example: right D-Pad arrow mapped to forward motion in Super Mario Bros.

The magnitude of the response of a parameter to a control over time is critical to the “feel” of that control

A framework for thinking about this time response is:
- Attack
- Decay
- Sustain
- Release

Model of Mario’s Horizontal Movement

max speed

button pressed

<table>
<thead>
<tr>
<th>time (sec)</th>
<th>1 sec</th>
<th>2 sec</th>
</tr>
</thead>
</table>

button released
**ADSR**

- **Attack**
  - The controlled parameter ramps up to its maximum value

- **Decay**
  - Any automatic decrease from the maximum value before reaching a steady-state value of the parameter
  - Often not present in video games
  - When present, is often a mistake

- **Sustain**
  - A steady-state value for the parameter, often maintained for as long as the player uses the control

- **Release**
  - The controlled parameter decreases back to some background state (down to no activity, no firing, etc.)

- Each of these are variables you can manipulate in creating your game’s feel
Immediate Response

- Immediate response to control
- Very short attack/release
- Example: player in Donkey Kong
- Responsive, but stiff
- Good for precise control: movement and jumping
Floaty Controls

- A longer attack phase results in a floaty, loose feel
- Sometimes floaty is good (Asteroids), but often is not
- Causes problems when players feel there is no immediate response to input

![Diagram showing time (sec) and max speed with control pressed and released]
Tight, responsive

- A short attack phase makes controls feel tight and responsive
Game states

- It is often the case that a player avatar (or the overall game) will have multiple distinct states
  - Example: Mario can be on the ground (state 1), or in the air (state 2)
- The ASDL response curve can vary depending on state
  - Strength of response to left/right controls is much smaller when Mario is in the air
  - This gives you an extra degree of freedom when designing your controls
  - Two or more sets of responses mapped to the same input
Exercise #1

- Go to:
  - [http://www.game-feel.com/?page_id=8](http://www.game-feel.com/?page_id=8)
  - Download example: Demo CH13-1 (*not* -2)
  - With a few partners, play around with the various inputs until you get a Mario character that has good game feel
Exercise #2

- With your team, create one or more ADSR charts to describe the response of your player avatar
- Discuss whether this is the best ADSR curve for your game
- Be prepared to discuss this during team meetings