Today

• Reminder
• Computational prototypes
• Platformer innovation
Syllabus Reminder
Computational prototypes
Design questions

• How will a core spatial/control mechanic feel? (Interface-in prototyping)

• Will the emergent NPC/enemy behavior be what we expect? (AI prototyping)

• Are the permutations balanced relative to cost? (Unit customization testing)

• Good flow of narrative and space?

• Do the systems interact as expected? (Combos of resources, combat, other rules)

• Will the visual aesthetic be achievable and appropriate? (Design/tech border question)
Gingold on prototyping
Always Ask a Question

- Always ask a question, which will give you purpose, and have a hypothesis, which is a specific idea you are testing out.
- Example question: how do I control these fish with a mouse?
- Example hypothesis: flocking will make the fish move together, and every mouse click will drop an invisible “bomb” that will act as a repulser upon every fish’s steering AI...
Stay Falsifiable

• You must validate the results of your experiment

• Did your hypothesis work? Does your fish flock control scheme feel good to you? Do your friends find that it feels good? Does it work in the context of your game idea?

• You can never user test and play test an idea too early
Persuade and Inspire

• We’re making entertainment and art — your prototype should be cool, fun, and excite people
• If you and your peers are compelled, so will your players be
Work Fast

• Try to minimize time to your first “failure” (rejecting a hypothesis), and don’t be afraid to push the eject button
• Prototypes don’t need “engines”
• Prototypes are slipshod machines held together by bubble gum and leftover bits of wire that test and prove simple ideas as quickly as possible
Work Economically

• You’re making something small and beautiful, invest development effort wisely

• In order to work fast, you must stay small: don’t do too much at once, or you’ll never make progress

Carefully Decompose Problems

• Don’t bite off more than you have to at once. If you prototype all systems simultaneously you will fail — you can’t work fast, or reach any kind of conclusion.

• To build it all at once is to build the actual game, which is hard.

• You must be careful, because problems are sometimes connected in non-obvious ways, biting you later.
Computational prototypes

• For next week, at least show your core mechanic and world working together
• Answer additional questions if time permits
• You will demo prototypes in section
• You must turn them in by putting in the right folder, naming the right name — test it works on another computer!
• Questions?
Platformer innovation
Prince of Persia: The Sands of Time
Braid

3:30
Portal
Closure

http://www.newgrounds.com/portal/view/480006
Now out for PS3!
There's a commercial version for XBLA and Windows.
Lost in Shadow

Platforming on shadows
as a shadow — with a helper

2:30
Super Meat Boy

Side platforming — and difficulty from Santa Cruz!
Bit.Trip Runner

Rhythm-based platforming —
also from Santa Cruz!
Snapshot

Capture objects/creatures with snapshots
reposition to move, solve puzzles, etc
Fez

2D platforming in a rotatable world
Dorkly’s take

Indie games are in dialogue with game history — like indie directors with the movies of their childhood
Changing the platform fundamentals

- Changing space's connections (*Portal*)
- Changing space's presence (*Closure*)
- Changing space's objects (*Spelunky*)
- Changing space's dimensions (*Fez*)
- Moving objects non-traditionally (*Snapshot*)
- Moving player character non-traditionally (*Super Meat Boy* & *Bit.Trip Runner*)
- Changing platformer time (*PoP* & *Braid*)

If you’re making a platformer, something to consider
Time isn’t just for platformers

And “rewindable time” is just one alternate operational logic for time
Cursor *10

http://www.nekogames.jp/mt/2008/01/cursor10.htm
Timebot

http://www.piratejuice.com/games/timebot/
The Misadventures of P.B. Winterbottom

http://www.winterbottomgame.com
These three games

• What do we notice about them?
• You “collaborate with yourself” in all three, over time
• But the types of challenges and specifics of collaboration/time mechanic are different
These three games

- *Cursor* *10* focuses on mouse movement & linear progression in an abstract world
- *Timebot* focuses on logical door & trigger puzzles w/ overlapping time in world with robots, gears, energy levels, time travel, etc
- *Winterbottom* focuses on platforming challenges (getting to locations, gathering objects) and direct interaction with past versions in developed silent-film world
Achron: Time travel RTS

What if you collaborated with yourself against someone collaborating with herself?
Upcoming

- Computational prototypes and schedules due this week