cisc3665
game design
fall 2011
lecture # I.1
introduction

topics:

- introduction to the course
- introduction to game design
  - aspects of game design
  - the game experience
  - what is a game; genre
  - game elements; mechanics
  - game development
- to do

instructor:

- Prof Elizabeth Sklar, *sklar@sci.brooklyn.cuny.edu*, AIM screen name: *agentprof*

course web page:

- http://www.sci.brooklyn.cuny.edu/~sklar/cisc3665

---

course content

- course description:
  - Introduction to designing the intelligence behind computer games.
  - Fundamentals of designing, programming and troubleshooting game behavior.
  - Documenting and critiquing design.
- prerequisite:
  - CISC 3120 or CISC 3110
- requirements:
  - readings (provided in class — no textbook!)
  - USB Flash drive (for labs)
  - attendance
  - assignments
  - term project
  - exams

---

topics

- the following topics will be covered in 5 curricular units:

  I. Introduction to Game Design and the Processing Environment

  II. Introduction to Game AI and Agents

  III. Game Physics, Motion and Perception

  IV. Nuts & Bolts: Scripting, Documentation, Data Collection and Analysis

  V. Game Theory and Behaviors

---

sources

- AI for Game Developers, by David M. Bourg and Glenn Seemann. O'Reilly Media, 2004.
- Programming Game AI by Example, by Mat Buckland. Worldware Publishing, 2005.
- Machine Learning, by Tom M. Mitchell. McGraw-Hill, 1997.
- Gameplay and Design, by Kevin Oxland. Addison-Wesley / Pearson Education, 2004.
- Processing: A Programming Handbook for Visual Designers and Artists, by Casey Reas and Ben Fry. MIT Press, 2007.
- Getting Started with Processing, by Casey Reas and Ben Fry. O'Reilly Media, 2010.
- On Game Design, by Andrew Rollings and Ernest Adams. New Riders Publishing / Pearson Education, 2003.
- Game Design: Theory & Practice (2nd Edition), by Richard Rouse III. Wordware Publishing, Inc., 2005.
- The Art of Game Design: A Book of Lenses, by Jesse Schell. Morgan Kaufmann. 2008.
- An Introduction to MultiAgent Systems (2nd edition), by Michael Wooldridge. Wiley, 2009.

course structure

- **5 units**
- each unit has lectures and labs
- the labs will be hands-on sessions using computers in 5301 N
- the assignments will be:
  - written assignments
  - design documents
  - software prototypes
- late policy

- your grade =

written assignments	20%
lab assignments	20%
term project	15%
midterm exam	15%
final exam	30%

about me

- undergrad: Barnard, CS major/English minor, BA 1985
- 10 years of industry experience working as a scientific and business programmer
  - MIT Lincoln Laboratory (http://www.ll.mit.edu/)
- grad school: Brandeis University, MA 1997, PhD 2000
- previous teaching:
  - Monash University, Melbourne, Australia (Summer 2000)
  - University of Melbourne, Melbourne, Australia (Summer 2000)
  - Boston College, Massachusetts (Fall 2000–Spring 2001)
  - Columbia University (Fall 2001–Spring 2005)
  - Brooklyn College, Fall 2005– ...
- research area: *artificial intelligence and multi-agent systems*
  - agent-based modeling of human behavior (e.g., in games)
  - human/multi-robot teams
  - data mining and visualization

aspects of game design [schell]

- animation : "to give life" to characters in your game
- anthropology : "study your audience in their natural habitat"
- architecture : understand relationships between people and spaces
- brainstorming : lots of new ideas!
- business : game *industry* → make games that make money
- cinematography : some games have bits of movies in them
- communication : work with team; interview audience
- creative writing : narrative of your fictional world
- economics : your fictional world may have its own economy
- engineering : main focus of this class—how to make it all work
- history : your game may be set in a historical setting
- management : you may need to manage the develpment team
- mathematics : goes along with engineering; probability, graphics, etc

- music : enjoyable background music, ambiance
- psychology : make your audience happy
- public speaking : talk about your game, presentation
- sound design : fx, other than music
- technical writing : documentation!
- visual arts : graphic elements

the game "experience"

- experience: "the game *enables* the experience" [schell, p10]
- cautions:
  - "peril #1: introspection can lead to false conclusions about reality" [schell, p15]
  - "peril #2: what is true of my experiences may not be true for others" [schell, p16]
- responses:
  - dissect your experiences, thoughts and feelings
  - observe and analyze, but not too much → "analysis paralysis"
  - analyze memories
  - experience something once, then go back a second time and analyze it
  - observation: quick glances or long study; but silently (watch, don't talk)
  - determine and capture the essence of the experience
  - reality is what you feel

---

the game experience: what players want and expect [rouse]

- players want:
  - a challenge
  - to socialize
  - a dynamic solitary experience
  - bragging rights
  - an emotional experience
  - to explore
  - to fantasize
  - to interact
- players expect:
  - a consistent world
  - to understand the game-world's bounds
  - reasonable solutions to work
  - direction

- to accomplish a task incrementally
- to be immersed
- some setbacks
- a fair chance
- to not need to repeat themselves
- to not get hopelessly stuck
- to do, not to watch

- "Players do not know what they want, but they know when it is missing." [rouse, p18]

---

the game experience: elements of gameplay [rouse]

- Unique Solutions
  - anticipatory vs complex systems
    * good designers try to guess what players of their game will do
    * you don't need to program in every possible response;
    * instead you can anticipate what players will (try to) do and respond to those things
  - emergence
    let player create a new solution; i.e., player-defined solution "emerges"
- Non-Linearity
  examples:
  - storytelling: non-linear story is tied to non-linear game play; where decisions made by player during the game change the outcome of the story and the game
  - multiple solutions: different players will solve problems in different ways; game should be able to handle multiple possible solutions to a problem
  - order: don't make everyone solve sub-steps in the same way
  - selection: allow players to select which challenge to solve next

- Modeling Reality
  - consider how much reality is necessary
  - decide how much reality is really necessary and stick to that much
- Teaching the Player
  - introduce complexity gradually
  - maybe provide tutorials
- Input/Output
  - controls and input:
    * make sure players know what the game controls are and how to use them
    * don't put in too much
    * importance of user-interface design in designing game controls
  - output and game-world feedback:
    * players should know and understand where/how changes in the game-world come about
    * players should receive feedback for the things they do
    * keep things visual and intuitive

---

what is a game?

- a game is something you play
- games like "Draughts" (Checkers) date back to
  - ancient Iraq (3000 BC) and
  - Egypt (6000 BC; game called "Alquerque")
- games must have simple rules and boundaries: these must be clearly defined [rouse]
  - rule: device to control and govern the game
  - boundary: limitations of game; usually physical, i.e., space

---

- qualities of games [schell]
  1. games are entered willfully
  2. games have goals
  3. games have conflict
  4. games have rules
  5. games can be won and lost
  6. games are interactive
  7. games have challenge
  8. games can create their own internal value
  9. games engage players
  10. games are closed, formal systems (i.e., have boundaries).

---

game genres [oxland]

- Make-up
  - core genres:
    * sports
    * adventure
    * action
    * simulation
    * strategy
    * puzzle
    * role-play
    * management
    * uncategorized
- Theme
  - e.g., role-playing game (RPG) with wizards or real-time strategy (RTS) war game
- Structure
  - e.g., two halves of a soccer game; laps in a racing game

- Game mechanics
  - e.g., moving parts
  - types of challenges in the game
  - interactive elements

genre: Role-playing game (RPG)

- intially MUD (Multi User Dungeon): board games, like "Dungeons and Dragons"
- MMORPG (Massively Multiplayer On-line Role Playing Games): text-based, like "Achaea" (`http://www.achaea.com/`)
- vs Second Life (`http://secondlife.com/`) — is this an RPG/MMORPG?
- player creates/instantiates (everything) about their character
- components:
  - character creation
  - character evolution
  - character class/type, attributes and skills
  - inventory management (stuff player can have/carry)
  - melee (interaction with other players, often in combat)
  - quests
  - interactive story
  - adventure

genre: Real-time strategy (RTS)

- war games, e.g., "Command and Conquer" (`http://www.commandandconquer.com/`), "World of Warcraft" (`http://us.battle.net/wow/en/`)
- components:
  - harvesting resources
  - building a community (generally a home base or bases)
  - spawning units with varying capabilities
  - strategic attacks
  - destroying the opponent

genre: Sports games

- one of the most popular genres
- there are action games (e.g., soccer, baseball) and extreme games (e.g., snowboarding, skateboarding)
- all games within a sport (e.g., soccer) have the same rules; but the implementation differs!
  - visual presentation
  - input/user controls
  - sound
- allow little (or no) room for creativity in rule changing on the part of the player
- components:
  - competition
  - performance challenges
  - quick reflexes

genre: Adventure games

- player embarks on an adventure (like RPG)
- requires player to think (like RTS)
- interactive storylines are revealed as player goes on adventure
- defined by: challenging situations and puzzle-solving that reveal story and move the game forward
- today's sub-genres: "action adventure", "survival horror"
- components:
  - puzzle-solving challenges
  - player has to think
  - intrinsic interactive storyline tied to challenges, revealed as challenges are overcome
  - multi-threaded plot
  - adventure journey from one location to another
  - a central protagonist

genre: Action games

- historically: big arcade games
- not much thought required; player must *react*; i.e., rely on player's quick reflexes, not brain
- quick to play (i.e., episodic); doesn't go on for days, like adventure games or RPGs
- components:
  - action
  - quick reflex movements
  - little thought process required
  - pick up and play (not over days)
  - no steep learning curve

genre: Simulation games

- simulation of some real-world activity, e.g., flying a plane
- components:
  - real physics
  - real vehicles
  - steep learning curve

genre: Puzzle games

- contains a problem that is not easily solved
- e.g., Tetris (or is this an action game??), TextTwist, Scrabble?
- puzzle should challenge the player
- often embedded in an adventure game

genre: Management games

- allows player to construct cities or households
- e.g., SimCity (or is this a simulation game??)
- have economic and social elements

game elements [oxland]

- Feedback: game's response to player's action
  - straightforward in Checkers
  - needs to be programmed into electronic games, e.g., Tetris
  - types:
    * visual
    * audio
    * action
    * "NPC" (non-player controlled characters)
    * accumulative (e.g., score)
    * emotional (provoke player)
    * fulfillment (game should stimulate this)
    * informative (player should know what the feedback is and what it means)
      "Action-Reaction-Feedback = Cause and effect" [oxland, p16]

- Interface:
  - GUI: graphical user interface
  - provides for input control/input and gives feedback
  - should be intuitive, friendly
- Context sensitivity:
  - in a complex domain, you may want the system to automatically (autonomously) handle some of the more complex aspects of control
  - e.g., flight simulator, formula-one car racing – the actual pilot or race car driver will have some context-sensitive computerized system components to help make their jobs easier
  - you can build the same types of controls into a game, and game player will still have a good experience
- Goals, quests and challenges:
  - define a core objective
  - define sub-goals for player to achieve on the way to achieving the core objective

- Environment design:
  - environment = game-world
  - are there levels?
  - define balance: to keep the game manageable, grounded in reality
  - define progression: to keep things interesting for the player ⇒ continuous challenge
- Balance:
  - speaks to the development process
  - balance gameplay with other aspects of development
  - get a playable prototype up and running and play; then iterate back through design/code/test to improve game
  - "playability needs to begin as soon as possible" [oxland, p22]
  - e.g., use stick figures, boxes, etc., to create a playable prototype before complete graphics are ready

game elements [schell]

1. mechanics: procedures and rules of the game

2. story: sequence of events that unfold in the game

3. aesthetics: how the games looks, feels and sounds

4. technology: how the game is implemented (i.e., hardware)

game mechanics [schell]

- mechanic 1. space
  - discrete or continuous
  - number of dimensions
  - boundaries of the game space
  - subarea(s): nested or not, connected or not
  - are there multiple ways to model the space?
- mechanic 2. objects, attributes and states
  - each object has attributes
  - each attribute has a state
  - accessibility of attributes and states: to character, to player, other players, other characters
- mechanic 3. actions
  - operative actions: things a player can do
  - resultant actions: results of operative actions

- mechanic 4. rules
  - operational rules: what the player can and cannot do
  - foundational rules: underlying formal structure of the game
    e.g., "player rolls a die" (operational) vs "player gets a random number between 1 and 6" (foundational)
  - behavioral rules: how the player should behave, i.e., good sportsmanship; code of conduct
  - written rules: the rules that come with the game
  - laws: define a tournament structure
  - official rules: merging of written rules with laws ... kind of a vague definition
  - advisory rules: about strategy
  - house rules: things people do customarily that aren't written down (e.g., "free parking" in monopoly)
  - object of the game
    * The most important rule!
    ∗ must be concrete
    ∗ must be achievable
    ∗ should be rewarding

- mechanic 5. skill
  skills of a player; i.e., how a player uses the rules to do well in the game
  - physical skill, e.g., tennis
  - mental skill, e.g., chess
  - social skill, e.g., scherades
- mechanic 6. chance
  games of chance, with some stochastic (random) element

game development: loop [schell]

- step 1. think of an idea
- step 2. try it out
- step 3. keep changing and testing it until it seems good enough
- *the more times you iterate through the develop/test loop, the better your game will be!*

game development: schell's eight filters

1. artistic impluse: "does this game feel right?"
2. demographics: "will the intended audience like this game enough?"
3. design of player's experience: "is this a well-designed game?"
4. innovation: "is this game novel enough?"
5. business and marketing: "will this game sell?"
6. engineering: "is it technically possible to build this game?"
7. social/community: "does this game meet our social and community goals?"
8. playtesting: "do the playtesters enjoy the game enough?"

to do

- in class:
  - fill out pre-semester survey and give it to me before you leave today
- at home:
  - check out the class web page:
    http://www.sci.brooklyn.cuny.edu/~sklar/cisc3665
  - read the handout [Oxland ch 1]
  - get a USB flash drive if you don't already have one