



Implement Test Code

Project 5

Due date: Wednesday, September 29th



Introduction

- Third in a series of related projects
 - Will build towards working game
- Focuses on
 - Development of game objects
- Using Flash

IMGD 1001

2



Motivation

- At core of game are the rules
 - Such as rules on gameplay (ie- payoff matrices)
- More than that
 - Hit points
 - AI for computer-controlled objects
 - Obstacles
 - Interface objects ...
- Begin prototyping the game
 - Gain experience implementing and testing game logic

IMGD 1001

3



Objectives

- Implement ten active game objects
- Provide two global game options
- Document your objects and options
- Submit your prototype and documentation

IMGD 1001

4



Overview

- Work in same group
- Use the treatment from Project 3
- Use the art from Project 4
 - Intent is *not* to more art or design (but can add – art is not “frozen”).
- All effort on implementing a variety of objects
 - in Flash!
- Evaluated based on
 - object activity
 - object interactivity
 - user interactivity
 - AI/reactivity
- Options
- Documentation indicating flexible grading

IMGD 1001

5



Details (1 of 3)

- At least 10 Objects
 - Next project on *Level Design* (and finishing prototype) so consider choices
- Each should have somewhat unique behavior
 - More than a copy or sub-class of another
- As a whole, your objects will meet the following criteria:
(Specific criteria next slide)

IMGD 1001

6

Details (2 of 3)

- **Object Activity** - Change state, reflected to the user in some fashion.
 - Ex: change in location (motion)
 - Ex: change in appearance (damaged object)
- **Object Interactivity** - interaction with other objects (i.e. – at least one changes state)
 - Ex: collision between two objects causes rebound
 - Ex: collision between two and "pickup" other item
- **User Interactivity** - respond to user input
 - Ex: pressing arrow keys moves avatar
- **AI/Reactivity** – "intelligent" behavior in reacting to objects around it. Adapt as situation changes.
 - Ex: Object pursues hero once awake

Details (3 of 3)

- For testing, create 1+ Stages (levels) in Flash
 - NOT meant to be playable levels (that's next project)
 - Do not spend much time on the levels themselves
 - Rather, use to test your objects (grading will use to evaluate),
 - Use as many levels and as many copies as needed
- Write a short README (text file)
 - Describes the objects, behaviors, and which objects fill which criteria
 - Indicate how the objects should be tested for grading!
 - Note global options and how affect gameplay
 - Indicate flexibility for grading!

Grading Guidelines

Criteria	Weight
Object Activity	15%
Object Interactivity	15%
User Interactivity	15%
AI/Reactivity	15%
Flexible	24%
■ Distribute across above criteria	
Game Options	10%
Documentation	6%

Submission

- Turnin (see Web page for instructions)
- Flash source and project files (.fla, .html, and .swf)
 - Will have art embedded
 - Can make separate .swf for each object behavior (but still turn in code)
- Documentation

Group Exercise

- Break into groups:
 - Blinky, Pinky, Inky, Clyde, Pac
- Consider objects in Pac-Man
- List and describe (5-7 minutes)
 - Object activity
 - Object interactivity
 - User interactivity
 - AI/Reactivity
- Are some objects related to others? If so, how?

