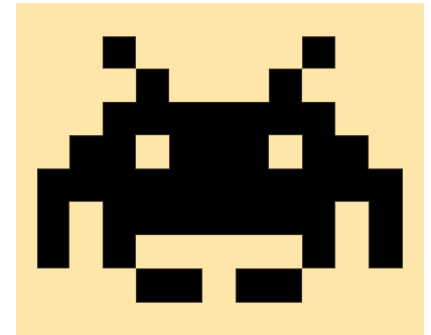


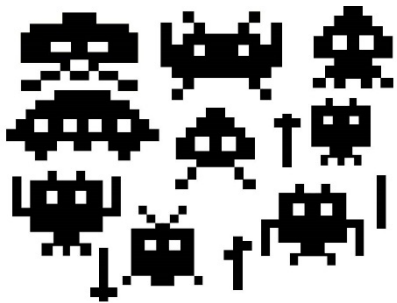


# CSCI 1106

## Lecture 07



## High Level Game Design



# Announcements

- Today's Topics
  - The Game Design Project
  - High-level game design
    - The Unifying Theme
    - The Game Story
    - Game Mechanics

# Your Mission: Write a Video Game

- Come up with an idea for a game
- Design the game
- Implement the game in Scratch
  - If it is not a first-person action game, be sure to clear it with the instructor first!
- Test and polish your game
- Create a user manual for the game
- Create a technical manual for the game

# Design Considerations

- What is the theme and objective of the game?
- How will the player move?
- How will the player win and lose?
- How will the player know how well they are doing?
- How will additional levels differ?
- How will you communicate the purpose, rules, and controls of the game?

# Project Evaluation

F	No game.
D	Sprites are moving on the stage, some interaction between sprites but game is not playable.
C-	Player movement is successfully added to the game.
C	The game tracks collisions between the player and game objects and responds to them.
C+	The game has a clear objective (including the use of positive and/or negative game objects).
B-	The player is able to easily track their progress through the game (in the form of points or some other measure / approach).
B	Winning and losing the game are both possible and the game rules, purpose and how to play are clearly communicated to the player.
B+	The game includes multiple levels that increase in difficulty.
A-	The game has some polish (looks OK) and includes audio effects.
A	The game is polished (looks good) and has some interesting special effects.
A+	The game is highly polished (looks really good) and is compelling.

To achieve a certain grade you must also have all of the features of the lower grades.

# Deliverables

- The game: .zip file
- The user manual (3 pages)
- The technical manual (7 pages)
- Presentation of your game during the presentation period
  
- The .zip file must be submitted via [prof1106@cs.dal.ca](mailto:prof1106@cs.dal.ca) of **February 13** before your presentation period
- All deliverables are due
  - On **February 24**, 2014
  - In hard copy (in class)

# The User and Technical Manuals

## User Manual

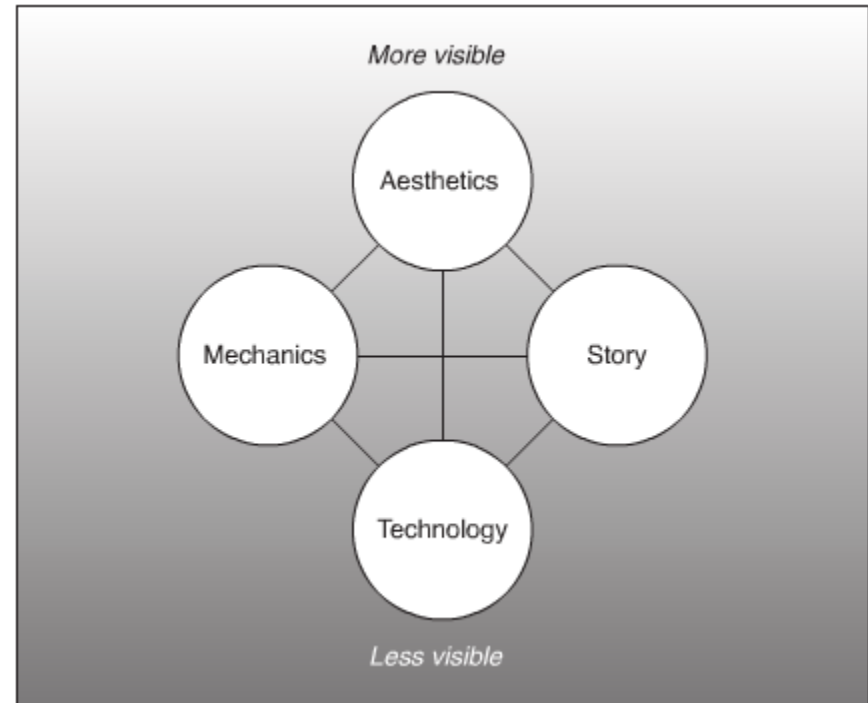
- Contents:
  - Title page with screenshot
  - Game overview
  - Rules
  - How to play
- 3 pages, 11pt font
- Worth 20% of the written component
- Templates are available on course website
- Rubric is in the project specification

## Technical Manual

- Contents
  - Title and Authors
  - Introduction
  - Description of Concept
  - Description of Sprites
  - Description of Stage
  - High-level Description
  - Description of Important Scripts
  - Description of Artwork and Sound
  - Future Work
- 7 pages, 11pt font
- 80% of the written component

# High-Level Game Design

- Game Elements
  - Story
  - Mechanics
  - Technology
  - Aesthetics



- Idea: The elements work together to create a *unifying theme* in the game





# The Game Story

- There's nothing like a good story to pull you in...
- A story is composed of:
  - A "world"
    - A place with consistent properties
    - e.g. physics, magic, culture, etc.
  - Characters
    - Individuals with likes/dislikes, personalities, and goals
    - Stock Characters: e.g. soldiers, clerics, plumbers
  - A quest
    - Why are we/they here?
- The story immerses the player
  - Transports them into the "world"
  - Whets the interest of the player (first 100 pages)
- Separates great games from ok games

# Story Considerations

- Depth
  - How detailed or grand is the story to be?
  - Epic? (Star Wars)
  - Simple backstory? (Angry Birds)
- Delivery
  - How is the story communicated to the player?
  - Prologue? Snippets? Chapters?
  - Does the player choose the direction of the story?
- Pacing
  - Rate of story telling corresponds to speed of the game

# Game Mechanics

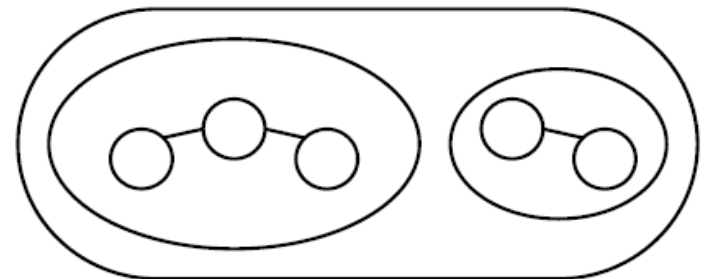
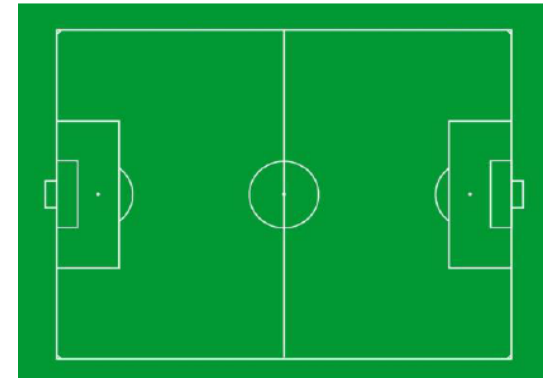
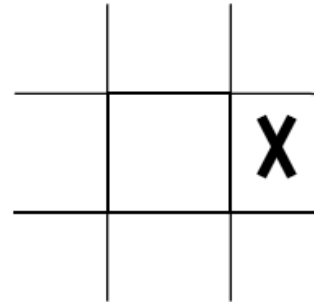
- Idea: Use game mechanics to
  - Implement the game story
  - Support the unifying theme of the game
- Game mechanics comprise
  - Rules
  - Environment
  - Actions
  - Chance (Randomness)
  - Skills

# Game Mechanics: Rules

- Written rules of play (what happens when I...)
  - User manual
  - Game code
- Unwritten rules
  - Etiquette
  - Sportsmanship
- Object of the game (how do I win the game)
  - Clear
  - Achievable
  - Rewarding/Fun

# Game Mechanics: Environment

- Spaces
  - Discrete or continuous?
  - Boundaries?
  - Nested Spaces?
- Number of players
  - Computer
  - Human
- Physics
  - Interaction of objects



# Game Mechanics: Actions

- Primitive Actions (private's view)
  - Moving the player
  - Shooting
- Strategic Actions (general's view)
  - Protecting a zone
  - Ambushing
- Most games require combination of both types of actions

# Game Mechanics: Chance

- Adds a surprising or unexpected elements
  - The so called "secret of fun"
- Consider how probabilities will factor into the play over the duration of the game
  - Power-ups
  - Density of projectiles
- Some predictability is useful! Why?
- The "chance trade-off"
  - A lot of randomness: game is about tactics, short term
  - A little randomness: game is about strategy, long term
  - Good games have the right mix



# Game Mechanics: Skills

- Idea: The right amount of challenge will keep the player interested
- Three types of skills:
  - Physical Skills
    - Strength, dexterity, coordination, and endurance
    - E.g. How fast can I hit that button?
  - Mental Skills
    - Memory, observation, and problem solving
    - E.g., The answer is ...
  - Social Skills
    - Reading and fooling opponents
    - Coordinating with teammates
- Many successful games combine skills from multiple categories

# Modeling Game Mechanics

- How do we depict what happens in our game?
- Need to model
  - Actions : human and computer
  - States of the characters and objects
  - Rules as a result of actions
- Idea: Use state transition diagrams
  - E.g., Mario eats a mushroom
  - E.g., Mario gets hit

# Game Genres

- Idea: A set of stock (standard) mechanics that are used by similar games is called *genre*
- Examples:
  - Card games
    - Take turns playing cards
    - Rules govern what the cards mean and who wins
  - Racing games
    - Drive a vehicle on a race course
    - Get across
  - First-person shoot-em up
- Right choice of genre supports the unifying theme