What Is Alice?

- A modern programming tool
  - 3-D graphics
  - 3-D models of objects
- Animation
  - Objects can be made to move around virtual world (a simulation or video game)
- Developed at Carnegie Mellon University
- At Duke – Use Alice in CompSci 4
The Power of Alice

• Automatically keeps track of 3-D objects
  – What objects are in the virtual world
  – Types of objects
  – Positions of objects in the world

Classes and Objects

• Classes
  – In Alice, classes are predefined as 3D models

• Objects
  – An object is an instance of a class
    • Class: Chicken
    • Objects: Chicken, Chicken2, Chicken3
Objects in Alice

• Objects already exist
• Objects have parts

Galleries of 3D Objects

• Sources of 3D objects
  – Local gallery – comes with Alice
  – Alice web gallery
Object Position

- Objects
  - Are positioned in 3D space
  - Have six degrees of freedom

Program an Object
Methods

• Built-in methods
• Write class methods
• Write world methods

Inheritance

• Dragon
• FlyingDragon
Example

- Princess on balcony needs to be rescued
- Other characters: dragon and knight

Example – parameters and events

- People are trapped in a burning building
- Select which person will be rescued
Parameters

- Types and number of parameters must match with arguments

Call

Events

- Default event

- Other events
Three Events

• The argument sent to parameters depends on which person is mouse clicked

<table>
<thead>
<tr>
<th>Events</th>
<th>create new event</th>
</tr>
</thead>
</table>

- **World**

  - **When** is clicked on `randomGirl3`
    - do `firetruck.savePerson whichFloor: burningBuilding.thirdFloor` `whichPerson = randomGirl3` `howFar = 3`

  - **When** is clicked on `randomGirl2`
    - do `firetruck.savePerson whichFloor: burningBuilding.secondFloor` `whichPerson = randomGirl2` `howFar = 2`

  - **When** is clicked on `randomGuy1`
    - do `firetruck.savePerson whichFloor: burningBuilding.firstFloor` `whichPerson = randomGuy1` `howFar = 1`

• Note - we positioned fire truck so distance from floor X is X meters (to floor 3 is 3 meters)

Example

- Zeus was a powerful god in Greek mythology. When Zeus was angry, he would shoot a thunderbolt out of the heavens to strike anyone who got in the way
- The user will choose the philosopher who will be the next target of Zeus’ anger.
Storyboard

• Possible design – method with Object parameter named *who*, for object clicked

  **Event**: an object is mouse-clicked

  **Event handler**: `shootBolt`

  **Parameter**: `who` – object clicked

  Do in order

  prepare to strike object that was clicked
  thunder plays and lightning strikes object clicked
  lightning is repositioned for next strike

  – The actions in storyboard are complex
  – Break actions into simpler steps using stepwise refinement

Implementation
Modify code – Use conditionals

- Only shoot philosophers
- Only shoot a philosopher if he has not already been shot – turn black when shot

Implementation with Conditionals
Looping structures

• Loop – simple

• Loop – complicated (this is a for)

Looping Structures (cont)

• While – may need a local variable
  declare variable: count = 0

While count < 5
  penguin turn right 1 revolution more...
  increment count by 1 more...
Example - while

Collections

- In some animations, several objects must perform the same actions
  - Example: marching band marching
- It is convenient to collect all objects into a group (collection)
  - Major benefit – write code for all the objects in the group (rather than separate code for each object)
Creating Lists

- In Alice, a list can be a list of numbers, or a list of objects, or a list of colors, etc.
- Let’s create a list of chickens

Create List (cont)

- Type in name
- Select type
- Select “make a list”
- Add chickens to list (click “new item” 4 times)
- Result is:
Programming with a List

• Can “iterate through a list”
  – Do something to each item in the list
    • In order (use “For all in order”)
    • All together (use “For all together”)

Example/Demo: Iteration in Order

For each chicken in order
  chicken says “hello”
For each chicken in order
  chicken turns its head and neck around
Example/Demo: Iteration Together

For all chicken together
   chicken says “hello”
For all chicken in together
   chicken turns its head and neck around

Example - Sort animals by height

Start

Put Tallest two in place …

All in place!
Arrays in Alice

• In Alice, array is a data structure to organize objects or information
• An array is not visible, it is a way of organizing
• But….
  – Alice has a 3D model to help you “see” the array

Example – Create a visualization of an array of people

• Add 5 people to the world
• Add an array visualization
• Not an array yet, must add people to the array

• Positions in array numbered starting with 0
Initialize array - Add Alice to Array in position 0

- Alice automatically moves to the 0 position!

Add Soldier to the Array

- Soldier moves automatically to position 1 (which is the 2cd position)!
Add RandomGuy, Skater and Rockette

- The array initialization is complete!

- Set isVisible for arrayVisualization to false
  - Array not seen

Setting elements in array

- Objects in an array are called elements

- Use “let” to set a position in an array

- Using “let”.
Why use Alice?

• There are very few women in computer science
  – Not uncommon to have 20% or fewer women in a course
  – Nationwide CompSci enrollments are down
    • Dot com burst
    • Outsourcing

Does Alice attract females?

• Build stories and interactive games
• Current two year study – several universities
• At Duke
  – CompSci 4 Spring 2005
    • 22 preregister, 30 enroll (12 female + 3 African Amer.)
  – CompSci 4 Fall 2005
    • 20 preregister, 31 enroll (17 female – one is African Amer.)
  – Advertised in school paper
    • picture of ice skater
    • Web site of animations
Duke Emerging Scholars in CS

• New program to try to attract women in CS
• Collaborative effort – 7 other universities
  – UW Madison, Purdue, Georgia Tech, Rutgers, Beloit, Duke, Loyola, UW Milwaukee
• At Duke
  – One-year program
  – Take CompSci 4, CompSci 6, CompSci 18S

Alice Software – is free!

• Runs on Mac and PC
• CompSci 4 web site
  www.cs.duke.edu/courses/fall05/cps004/rodger/
• Textbook available
  – Learning to Program with Alice by Dann, Cooper, and Pausch

• Download from web
  www.alice.org