Repetition

- In many kind of animations, especially simulation and games, some actions happen again and again
  - Example
    - Games where targets randomly appear and are caught or shot down, then appear elsewhere
- Actions are made to happen again and again by running an instruction or method more than once

Example

- Bunny sneaks into garden and wants to eat broccoli. Bunny needs to hop several times over to broccoli
Bunny.hop

- Makes bunny hop up and down, making a sound and traveling .8 meters total
- See code in book
- How do we get bunny to hop many times over to the broccoli?

One solution

- What is the problem with this solution?

Counted Loop

- A counted loop is an alternative way to write repetitive code
- Repeats instructions a counted number of times

Demo - Code to hop 6 times

- The loop instruction executes a definite number of times, specified by a count
- Using a loop instruction
  - Saves time
  - Is convenient, easy to change the count
  - Can use a function in place of the count (must return a number)
Infinity times….

• If “infinity times” is selected for a loop, loop will run until the program is shut down

| 1 time | 2 times | 5 times | 10 times | infinity times |

Example

• What happens if we make the other bunny hop up and down infinity times?

How do we fix this?

• How do we get both bunnies to move, one infinitely and one definitely?

More Complicated Loops

• It is possible to place a loop within another loop statement, this is nested loops

• Example in book: double ferris wheel

• NOTE: Be Very Careful when using infinite loop! If something goes forever, it doesn’t stop!
Demo - Ferris Wheel

nested loops

Review: What is a Variable?

• Property that can be changed using \textit{set}

Problem

• Given a cow that can randomly appear and disappear.
• Want to add a score to count the number of times user clicks on the cow.
• User gets specified amount of time to click (timer).
• Cow stops moving when time is up.
• User wins if a target number of clicks is achieved in the specified time. Cow tells user if they won or not.
• \textbf{Start classwork now and I’ll show how to do the score}

Solution

• Add a new 3D text object
  – will keep track of times cow clicked on
  – type in “score,” as the name of the 3D text object
  – Then change its text value to 0
Add Mutable Variables

• What does Score need to keep track of?
  – current value
  – final value
  – increment value

• Add three mutable class variables

Mutable Variables Added

• Three variables added
• Write method to initialize them
• Call to initialize

Increment Counter

• Add an event to increment score's value when mouse is clicked on cow

First, change score.value

• Drag value over and set to score.value
• Use math to increase by increment
Second, display the new text value

- Each 3D text has a text value
- Drag this field and set
- A world built-in function can be used to display the number value as a string

Almost Done…

- Add another 3D text to just say the word score
  - I named it scoreText
  - Then changed text value to “score”
- Add code to repeat until target is reached
- Cow appears at end

ClassWork

- Start with ClickACow.a2w
- Add a Score
- Add a Timer – similar to score
  - Start at high value (say 20)
  - Count down instead
- Game is over when Timer runs down
  - If Target score is reached – you win