CSCI 1106
Lecture 9
Buttons, Text, Lists, and Random Numbers
Announcements

• No class this Friday
• Today’s Topics
  – Buttons
  – Text
  – Lists
  – Random Numbers
Button State

• Buttons are sprites with commonly three (3) states
  – **Up** is the normal state of the button
  – **Over** is when the mouse is hovering on the button
  – **Down** is when the button is pressed

• Idea: For each of the three states the button can have a different look (costume)
Creating Buttons

• Create *sprite with three costumes*
  – *Up*
  – *Over*
  – *Down*

• Have sprite receive FRAME event
  – If the mouse is touching the button
    If clicked *[Down]* use Costume 3
    Otherwise *[Over]* use Costume 2
  – Otherwise *[Up]* use Costume 1

• Only change costumes if necessary!
• When should we actually execute action associated with button?

*when this sprite clicked*
Text

• It is useful for games to display text
  – Instructions
  – Player information (score, health, level, etc)
  – Dialogue

• There are two types of text that we can display
  – Static text, which does not change during the game
    • Instructions
    • Dialogue
  – Dynamic text, which changes as the game progresses
    • Player information
Static Text

• To create static text on the stage
  – Use sprite whose costume(s) contain text
  – Place sprite where you want to text to be displayed
• Switch the static text by switching costumes
• Pros:
  – Easy to do
  – Can use any tool to create and render the text
• Cons:
  – Text cannot be modified once program is running
Dynamic Text

• Three options for displaying dynamic text:
  – Variables
  – Say/Think blocks
  – Third party blocks
Dynamic Text: Variables

• **To Use:**
  – Create a variable
  – Drag the field displaying the variable to where you want to place it
  – Modify variable to change the text being displayed on stage
  – Hide / Show the variable as desired

• **Pros:**
  – Easy to use

• **Cons:**
  – Does not look good
Dynamic Text: Say/Think Blocks

• Use these blocks in your scripts when you wish a sprite to say or think something

• Pros:
  – Easy to use
  – Looks ok

• Cons:
  – Text is associated with a sprite
  – In many cases, the text is neither said nor thought
    • e.g., Player information
Dynamic Text: Third Party Blocks

• There are additional blocks, implemented by other people available on the web
  – You will need to find them on your own
    • (How to ... in scratch)

• Pros:
  – Look good

• Cons:
  – Have to find them yourself
  – In many cases they are specialized
Lists

• A list is contiguous sequence of elements
  – Used to store multiple pieces of information at once, e.g.,
    • numbers
    • strings

• Lists can be manipulated using operations:
  – *add* item to end of the list
  – *delete* \(i^{th}\) item of the list
  – *insert* item at location \(i\)
  – *replace* \(i^{th}\) location with another
  – *access* \(i^{th}\) location
  – *Check if list contains specific item*
  – *Show/Hide*
## List Operations

### Codon

<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>O</td>
<td>G</td>
<td>A</td>
<td>C</td>
<td>G</td>
<td>G</td>
<td></td>
</tr>
</tbody>
</table>

- **Think item 3 of Codon**: item 3 is 3
- **Add G to Codon**: Codon is now [A, O, G, A, C, G, G, G]
- **Replace item 2 of Codon with C**: Codon is now [A, C, G, A, C, G, G, G]
- **Delete 4 of Codon**: Codon is now [A, C, G, A, C, G, G]
Looping over a List

<table>
<thead>
<tr>
<th>Codon</th>
<th>A</th>
<th>U</th>
<th>G</th>
<th>A</th>
<th>C</th>
<th>G</th>
<th>G</th>
</tr>
</thead>
</table>

1 2 3 4 5 6 7 ...

- set idx to 1
- repeat length of Codon
  - say item idx of Codon for 2 secs
  - change idx by 1

A U G A C G G
A Random Aside

- **Idea:** Most systems have a pseudorandom source of values
  - The source is an infinite sequence of values
  - The values look random
  - Are sufficiently random for our purposes

- Each system is a little different, but all work similarly
  - Each system provides a Random function
  - The function returns a value chosen pseudorandomly from a fixed range
in Scratch

• Scratch has a function
• Returns a value in the range $\text{min} \leq n \leq \text{max}$
• Value is selected at random from a uniform distribution
• What does a uniform distribution mean?

![Graph showing uniform distribution](image)
A Random Code Example

• If you wanted to implement a coin toss, how would you do it?
Another Random Example

- How do we place an object at a random horizontal position on the stage?
  
  \[ y = 50; \]
  \[ x = ??? \]

```markdown
set y to 50
set x to pick random -240 to 240
```

min value max value