End papers…

• Mistakes, obviously, show us what needs improving. Without mistakes, how would we know what we had to work on?
  ~Peter McWilliams, Life 101

• Mistakes are the portals of discovery.
  ~James Joyce (1882 - 1941)

Announcements

• New labs with quizzes
  * Take the quiz as many times as you want to improve your score
  * Bring questions to office hours and drop-in labs
  • Drop-in labs on Wednesdays before lecture are very sparsely attended—2 to 3 people!
    - 10:30am to 12:30pm in MGH 430

Project 2A

• JavaScript Storyteller

Iteration Principles

Again, and again, and again

JavaScript

• We are now in a foreign land where you don’t speak the language
• You recognize some sign posts—html—but JavaScript is brand new

Looping back through some fundamental concepts

BROWSERS, HTML, & JAVASCRIPT
Why JavaScript?

- JavaScript is a programming language that many Web browsers can understand, or interpret.
- JavaScript can be written with a simple text editor like NotePad++ or TextWrangler, or Dreamweaver, and tested in a Web browser.
- A JavaScript program is a list of commands or statements that the browser runs to add features to an HTML document.

The goals of programming:

- Increase user interaction in three ways:
  - Accept user input.
  - Calculate using user input.
  - Display results.
- A user can invoke actions and/or enter data. Programmers call these actions and/or data user input, or simply input.
- The computer can calculate a number, create some text, etc.
- The results shown to the user are called output data, or simply output. Sometimes the output will change the graphical user interface (GUI) of the program, which changes what the user sees and can do with the program.

Running JavaScripts

- How the Web browser reads an HTML document that contains JavaScript:
  1. The web browser starts to read through the HTML document from the first character in the top left to the last character in the bottom right.
  2. As the web browser reads through the document, the browser builds a document tree from the HTML elements it encounters.
  3. If the browser encounters a script element, it executes the code that is specified by the src attribute. If there is no src attribute defined, the browser executes the code that is in the contents of the script element.
  4. If the script generates HTML output, this output is then added to the HTML document tree.

Play it again, Sam.
Definitions

• Iteration, or looping, is the process of repetition:
  ∗ looping through a series of statements to repeat them

Major Types of Iterations

• For loop
  ∗ Count up
  ∗ Count down
• While loop
  ∗ Count up
  ∗ Count down
• Do…While

FOR LOOPS

Repetition is good

The for Loop Basic Syntax

for(<initialization>; <continuation>; <next iteration>)
{
<statement list>
}

• Text that is not in metabrackets <> must be given literally
• The whole sequence of statements in the statement list is performed for each iteration
  ∗ Computer completes the whole statement sequence of the <statement list> before beginning the next iteration

The Iteration Variable

• Control specification: the three operations in the parentheses of the for loop
  ∗ Control the number of times the loop iterates
  ∗ by using an iteration variable (must be declared)

JavaScript Rules for for Loops (cont’d)

• The World-Famous Iteration
  ∗ JavaScript uses the same for loop statement as other programming languages, so thousands of loops with this structure are written every day:
  ```javascript
  for ( j = 0; j < n; j++) {...}
  ```
  ∗ Most frequently written for loop of all time
  ∗ Easy to see iteration count:
    ∗ Always n times
The Iteration Variable (cont'd)

• Example:

```java
for ( j = 1; j <= 3; j = j + 1 ) {
    <statement list>
}
```

• Here’s what happens:
  - The first operation is the <initialization>.
    - Sets the iteration variable's value for the first iteration of
      the loop. Done only once.
  - The next operation is <continuation>.
    - Test. If the test has a false outcome, the <statement list> is
      skipped.
    - If the test has a true outcome, the <statement list> is
      performed. When the statements are complete, the
      <next iteration> operation is performed.
    - Repeats with the continuation text, performs same
      sequence of steps.

Running through a for loop

<table>
<thead>
<tr>
<th>Operation</th>
<th>Operating Value</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>j = 1</td>
<td>j's value = 1</td>
<td>Initialize iteration</td>
</tr>
<tr>
<td>j &lt;= 3</td>
<td>true, j is less than 3</td>
<td>First &lt;continuation&gt; test, continue</td>
</tr>
<tr>
<td>j = j + 1</td>
<td>j's value = 2</td>
<td>First &lt;next iteration&gt; operation</td>
</tr>
<tr>
<td>j &lt;= 3</td>
<td>true, j is less than 3</td>
<td>Second &lt;continuation&gt; test, continue</td>
</tr>
<tr>
<td>j = j + 1</td>
<td>j's value = 3</td>
<td>Second &lt;continuation&gt; test, continue</td>
</tr>
<tr>
<td>j &lt;= 3</td>
<td>true, j is equal to 3</td>
<td>Third &lt;continuation&gt; test, continue</td>
</tr>
<tr>
<td>j = j + 1</td>
<td>j's value = 4</td>
<td>Third &lt;continuation&gt; test, continue</td>
</tr>
<tr>
<td>j &lt;= 3</td>
<td>true, j is greater than 3</td>
<td>Fourth &lt;continuation&gt; test, terminate</td>
</tr>
</tbody>
</table>

How a for Loop Works

• Consider a computation on declared variables j and text

```javascript
text = "She said ";
for ( j = 1; j <= 3; j = j + 1 )
{
    text = text + "Never! ";
}
alert(text);
```

How a for Loop Works

• Consider a computation on declared variables j and text

```javascript
text = "She said ";
for ( j = 1; j <= 3; j = j + 1 )
{
    text = text + "Never! ";
}
alert(text);
```

How a for Loop Works

• Consider a computation on declared variables j and text

```javascript
text = "She said ";
for ( j = 1; j <= 3; j = j + 1 )
{
    text = text + "Never! ";
}
alert(text);
```

How a for Loop Works

• Consider a computation on declared variables j and text

```javascript
text = "She said ";
for ( j = 1; j <= 3; j = j + 1 )
{
    text = text + "Never! ";
}
alert(text);
```
How a **for** Loop Works

- **Demo:**
  ```javascript
  text = "The two-year-old said ";
  for ( j = 1; j <= 3; j = j + 1 )
  {
      text = text + "No! ";
  }
  alert(text);
  ```

JavaScript Rules for **for** Loops

- **The Iteration Variable:** `j = 1`
  - Must be declared, and follow rules for variable identifiers
  - `i`, `j`, and `k` are the most common choices

- **The Starting Point**
  - Iteration can begin anywhere, including negative numbers

JavaScript Rules for **for** Loops

- **Continuation/Termination Test** `j <= 3`
  - Test is any expression resulting in a Boolean value (true/false)
  - Continuation must involve iteration variable to avoid infinite loop

- **Step Size** `j = j + 1`
  - Amount of change from one iteration to the next
  - Often called the increment or decrement
    - Increment: `j + 1`
    - Decrement: `j - 1`

Experiments with Flipping Coins

- To practice **for** loops, we experiment with flipping electronic coins
- We can use the function `randNum(2)`, which returns either 0 (tails) or 1 (heads)
- Set up an iteration in which our `randNum()` function is performed 100 times, and statistics gathered

Experiments with Flipping Coins (cont'd)

```javascript
<script type="text/javascript">
  var header=0, tail=0; //Counters
  var i=1; //Iteration variable
  while (i<=100) //While loop
  {
    if (randNum() == 1) //If heads
      ++header;
    else
      ++tail; //If tails
    alert("Heads: " + header + " and Tails: " + tail);
  }
</script>
</head>
</body>
</html>
```

Demonstration

- **Coin toss...**
Experiments with Flipping Coins (cont'd)

- i ranges from 0 to 99, so the loop iterates 100 times.
- Conditional statement checks and records the outcome of random number generation.
- When random number is 1, count of heads is increased by 1 (heads++);
- When random number is 0, count of tails is increased by 1 (tails++);

A Nested Loop

- To run several trials, consider the entire loop we just looked at as one Trial.
- Create another for loop containing this Trial unit, adding a couple of needed statements.
- We have a loop within a loop (nested loop) which causes the Trial loop (0-99) to run five times.

Experiments with Flipping Coins

A Nested Loop

- To run several trials, consider the entire loop we just looked at as one Trial.
- Create another for loop containing this Trial unit, adding a couple of needed statements.
- We have a loop within a loop (nested loop) which causes the Trial loop (0-99) to run five times.

Experiments with Flipping Coins

A Nested Loop

- To run several trials, consider the entire loop we just looked at as one Trial.
- Create another for loop containing this Trial unit, adding a couple of needed statements.
- We have a loop within a loop (nested loop) which causes the Trial loop (0-99) to run five times.

Demo...
Experiments with Flipping Coins (cont'd)

• A Diagram of Results
  • To show how far off a perfect 50-50 score a trial is, display with diagram
  • Compute the distance from 50-50 and show that number using asterisks

```javascript
var text = 'Trial ' + j + ':
for (i = 0; i < Math.abs(heads-50); i++) {
  text = text + '*';
}
text = text + '\n';
alert(text);
```

Quick Write

• Closed book
• Your own work
• Put away
  • Laptops
  • Cell phones
  • Notebooks
  • Books
  • Etc.
• Stop talking
• Eyes to yourself
• Raise your hand if you have a question