**Announcements**

**Tip of the Day:** In solving programming problems, look for analogous situations & morph that solution.

**Project 2A due Wednesday**

**Once is Not Enough**

*Repeating instructions is the source of great power in computing*

**Iteration**

“Iteration” is another term for “repeat”
- Iteration doesn’t suffer from the question of whether the first item is counted … in iteration it always is. (Use “repeat” and “iterate” interchangeably unless it matters.)
- Iterating is usually called “looping” in programming
- Programming languages have many kinds of statements to help program loops
- In JS we will use the `for` statement

**Anatomy of For**

The `for`-statement syntax:

```javascript
for (<initialize>; <continue test>; <next iteration>) {
  <statement list>
}
```

- `<initialize>` -- gives iteration variable its first value
- `<continue test>` -- this test is performed before starting each cycle of loop; if false, quit
- `<next iteration>` -- the change to the iteration variable after each cycle

**An Iteration**

Iterations can count ...

```javascript
var i, text = ""; // Initialize text to empty string
for (i = 1; i <= 5; i = i + 1) {
  text = text + "Iteration no.: " + i + "n";
}
alert(text);
```

**Iterations Control Actions**

Iterations can replicate other things...

```javascript
var i, text = "It's funny!", newLineInJS="n";
for (i = 1; i <= 3; i = i + 1) {
  text = text + " Ha!
";
}
alert(text);
```

It is possible to make it a lot funnier by changing the limit variable to, say, `i <= 1000`
Key Points of Loops

The most important features of loops:

- The starting value of the iteration variable
- The ending value of the iteration variable
- The amount the iteration variable changes

As explained in the book, it is possible to completely control these features by properly setting the "control trio," but programmers have gotten in the habit of writing a single kind of iteration: WFI

World Famous Iteration

To loop n times the WFI has this form

```
for (i=0; i<n; i++) {
  <statement list>
}
```

Advantages:

- Fast to type
- The number of iterations is the number after <
- 0-origin makes it handy for most computations

World Famous Iteration

```
for (i=0; i<n; i++) {
  <statement list>
}
```

"Off By 1" Error

The most common error when working with iterations is to miscount by 1

- Everyone makes this mistake
- A common place where the "off by 1" error matters is in how many times a loop loops
- The importance of the WFI is it tells exactly

```
for (i=0; i<n; i++) {
  <statement list>
}
```

Number of iterations

```
for (i=0; i<n; i++) {
  document.write("[ " + i + " ]");
}
```

Doubly Nested Loops

A loop within a loop repeats repetitions

We know about names with multiple instances: Rocky 3, QE 2, John Paul 2

- The number is called the name’s index
- The least index is called the index origin
- In programming, variables that can be indexed are called arrays
- Declare arrays in JavaScript:
  ```
  var <identifier> = new Array (<num elements>);
  ```
- JavaScript arrays are 0-origin
- Reference array elements w/ brackets: A[0]
Loops and arrays work together

- Declare an array and initialize elements to 8
  ```javascript
  let j, A = new Array(5);
  for (j=0; j<5; j++) {
    A[j] = 8;
  }
  ```

  WFI and array’s indices both start at 0. Notice what would change to have 1000 elements — arrays and loops give power.

Summary

Iteration is very powerful because a small amount of code specifies a lot of computation:

- `for` gives full range of looping limits, steps
- Use any form of `for` that works, but using the WFI is a good habit to adopt
- In a doubly nested loop one iteration has another iteration as its `statement list`
- Arrays are variables with many elements that are referred to by their index.