Resume

- Placed our discussion in a historical context
- Talked about physical structure of the computer: hardware, CPU, storage
- Representing information using 0s and 1s
  - audio, images, text, numbers
- Sharing information among computers

Preview

- Writing computer programs
  - Basic programming skills
- Algorithm design
- Software distribution policies
- Social aspects of computing
- “Intelligent” computers?
- What computers cannot do

Programming 1

What is a computer program?

- "A sequence of instructions that a computer can interpret and execute“

— *WordNet 2.0*

What is a computer program?

- "A sequence of instructions that a computer can interpret and execute"

Is the following a computer program?

```
"Analyze the US Census!!!"
```

Answer: No, because it cannot be understood and therefore cannot be executed by a computer. It is an instruction written in English.

What is a computer program?

- "A sequence of instructions that a computer can interpret and execute“

Is the following a computer program?

```
Step 1. Peel 4 potatoes.
Step 2. Bring 4 cups of water to a boil.
Step 3. Add the peeled potatoes to the boiling water.
Step 4. Boil the potatoes for 10-15 minutes.
Step 5. Mash potatoes with a fork.
Step 6. Add milk, butter and salt.
```

Answer: No, because it cannot be understood and therefore cannot be executed by a computer. It is a sequence of instructions written in English.

What is a computer program?

- Computers don't understand English, French, Japanese, etc.
- Computer programs are written in *computer programming languages*; ie. C, C++, Java, JavaScript
- Provide the syntax (set of allowed words) and grammar (rules for using these words) for instructions

Example of a computer program

```
document.writeln("Hello, World!")
```

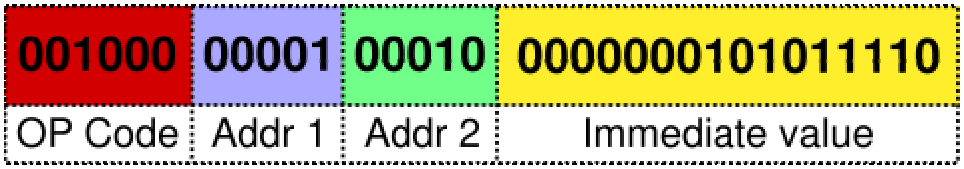
- Prints out “Hello, World” to the screen

- List of “Hello, world” programs in *many* different languages (general interest only):
- http://en.wikipedia.org/wiki/List_of_hello_world_programs

Machine Instructions (Hardware Level)

Example:

MIPS32 Add Immediate Instruction



Add the contents of the *register* r2 and 350 and store the result in the *register* r1

- Other kinds of instructions include:
  - Transferring data between registers or memory locations
  - Arithmetic or logical operations (use the ALU)
  - Control: test contents of a register and jump to a location

This is all a computer does!!!

High-level to Low-level

- Computer programs written in a high-level language, like Javascript, are translated into machine instructions
- High-level "source code" is easily human-readable
- These machine instructions are not:

```
00101010010001110110111110010101
00010000001001101011011111011011
10011101000111001001100101011100
00101101100001000010000011010000
```

JavaScript

- Used to add functionality to websites
- May be executed using any web browser
  - JavaScript may be disabled in your browser, follow the steps in http://www.iatn.net/help/javascript/ to turn it on
- Actually embedded in HTML:

```
<html>
<head><title>My first JS Program</title></head>
<body>
<script type="text/javascript">
document.writeln("Hello, World!")
</script>
</body>
</html>
```

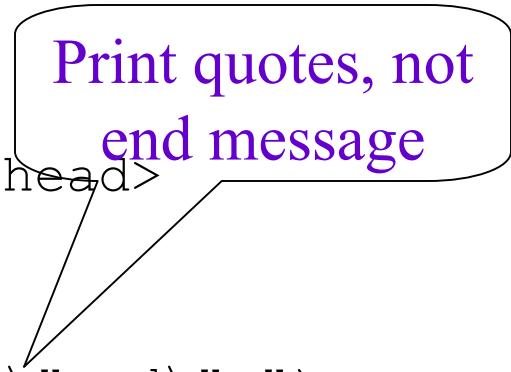
Steps for writing JavaScript programs:

1. Create a new text file
2. Type your code in the text file (including the HTML)
3. Save your file with an .html extension (e.g., file.html)
4. Open your file in a web browser to make sure that your code works

JavaScript and HTML

- The output of all programs will become part of the HTML text. Therefore, our programs can output hypertext and the browser will render all the text before showing it on the screen.

```
<html>
<head><title>Text Formatting</title></head>
<body>
<script type="text/javascript">
document.writeln("<font size=13 color=\"red\">")
document.writeln("This text is big and
  red</font>")
</script>
</body>
</html>
```



JavaScript and HTML

```
<html>
<head><title>Text Formatting</title></head>
<body>
<script type="text/javascript">
document.writeln("<font size=13 color=\"red\">")
document.writeln("This text is big and red</font>")
</script>
</body>
</html>
```

```
<html>
<head><title>Text Formatting</title></head>
<body>
<font size=13 color="red">This text is big and red</font>
</body>
</html>
```

Comments

- *Comments* are ways to annotate your code. They are ignored by the JavaScript intepreter.
- Everything to the right of "//" is considered a comment

```
// My favorite JS program
// Student # 119929192
// Name: Bobby
// I can put anything here! YAAAAAAAAAAAAAAAAAYYYYY!
document.writeln("GO HABS GO!! <br>"); // Even here!
document.writeln("Life is good.");
```

- This program only has 2 JS statements.

Variables

- *Variables* are containers (in memory) for information that you want to store.
- Announcing the existence of a variable is called *declaring* the variable.
- Putting a value in the container is called *assigning* the variable.
- You can print out the values in the container using the print statement without any quotations. What happens if you use quotations?

```
var x // a declaration
x = 3 // assigning the value of 3 to x
document.writeln(x) // writes the *value
  // of x* to the screen
```

Variables

- Variables can be *reassigned* values at different points in the execution of the problem.
- When a variable is reassigned a value, the old value is lost.

```
var x = 4                  // a declaration and assignment in one
var y = 8                  // another declaration and assignment
document.writeln(x) // prints 4 to the screen
document.writeln(y) // prints 8 to the screen
x = 6                      // re-assigning x
document.writeln(x) // prints 6 to the screen
document.writeln(y) // prints 8 to the screen
```

Arithmetic Operations

- The arithmetic operators (+, -, *, /) are for mathematical operations:

```
var x = 2*2                 // '2*2' is an expression that evaluates to 4,
                            // so x is assigned the value of 4
var y = 8                   // another declaration and assignment
var z = x + 2               // 'x + 2' is an expression that evaluates to 6
                            // so z is assigned the value of 6
y = y - z                   // 'y - z' evaluates to 8 - 6, which is 2
                            // so y is re-assied to 2
document.writeln(x+y+z) // 'x+y+z' evaluates to 12, so 12 is printed
```

JavaScript Lab

- Who is interested?
- What day/time?