

# Computer Science E-1

Understanding Computers and the Internet

## Lecture 11: Programming

Thursday, 8 December 2005

David J. Malan  
malan@post.harvard.edu

1

## Agenda

- Pseudocode
- Constructs
  - Instructions
  - Variables
  - Conditions
  - Branches
  - Loops
- Languages
  - Interpreted
  - Compiled
- JavaScript

2

# Pseudocode



3

# Pseudocode

## **Self-Counting Algorithm for Students**

1. Stand up.
2. Assign yourself the number 1.
3. Find someone else that is standing up. (If no one is standing, remain standing until I call on you.)
4. Add your number to that person's number; the total is your new number.
5. One of you should then sit down—decide who sits however you want.
6. The person still standing should now go back to step 3.

4

# Pseudocode

## Diaper-Changing Algorithm for (Clueless) Dads

1. Place Baby on a flat surface.
2. Unfold the Pampers and fan out the end that the tapes are on.
3. Grasp Baby's ankles by thumb and middle finger, inserting index finger between his ankles to keep from pressing them together.
4. Raise legs and hips off the table.
5. Slide the fanned-out end with the tape under Baby.
6. Bring diaper up between legs, extra thickness in front for boys, in back for girls if they generally lie on their backs.

5

# Constructs

## Sock-Donning Algorithm

1. let socks\_on\_feet = 0
2. while socks\_on\_feet != 2
3.     open sock drawer
4.     look for sock
5.     if you find a sock then
6.         put on sock
7.         socks\_on\_feet++
8.         look for matching sock
9.         if you find a matching sock then
10.             put on matching sock
11.             socks\_on\_feet++
12.             close sock drawer
13.         else
14.             remove first sock from foot
15.             socks\_on\_feet--
16.     else
17.         do laundry and replenish sock drawer

6

# Languages

## Interpreted

```
<!DOCTYPE html
  PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "DTD/xhtml11-transitional.dtd">

<html>
  <head>
    <title>Hello, World!</title>
  </head>
  <body>
    Hello, World!
  </body>
</html>
```

7

# Languages

## Compiled

```
int
main(int argc, char * argv[])
{
    printf("Hello world!");
    exit(0);
}
```



```
10000011 00000001 00010001 00000000 00111101 11111100 01110100 00111101
00000000 01000000 00000000 00000000 00000000 00000000 00000000 00000000
10010000 00000000 00000000 00000000 01010000 00000000 00000111 00110000
00001011 00000001 00001011 00000011 00001010 00000000 00000000 00000000
00000000 00100000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00100000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
01110000 00010000 00000000 00100000 00000001 00000000 00000000 00000000
00000000 00000000 00000000 00100000 00000001 00000000 00000000 00000000
00000000 00000000 00000000 01000000 00000001 00000000 00000000 00000000
00000000 00100000 00000000 01000000 00000001 00000000 00000000 00000000
11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
10010000 10000000 00000000 01000000 00000001 00000000 00000000 00000000
00101110 01100100 01111001 01101110 01100001 01101101 01101001 01100011
10110000 00000100 00000000 00100000 00000001 00000000 00000000 00000000
10110000 00000100 00000000 00100000 00000001 00000000 00000000 00000000
10100000 00000001 00000000 00000000 00000000 00000000 00000000 00000000
10110000 00000100 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00100000 00000000 00000000
[...]
```

Source from <http://www.fas.harvard.edu/~cscie1/distribution/lectures/11/compiled/hello.c>.

8

# Languages

## Compiled

```
#include <iostream>
```

```
int
```

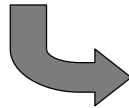
```
main(int argc, char * argv[])
```

```
{
```

```
    cout << "Hello world!" << endl;
```

```
    exit(0);
```

```
}
```



```
10000011 00000001 00010010 00000000 01100001 11111100 01110100 00111101
00000000 11000000 00000010 00000000 00000000 00000000 00000000 00000000
10010000 00000000 00000000 00000000 01010000 00000000 00000111 00110000
00001011 00000001 00001011 00000011 00001010 00000000 00000000 00000000
00000000 00100000 00000010 00000000 00000000 00000000 00000000 00000000
00000000 10100000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00010000 10110000 00000000 00100000 00000001 00000000 00000000 00000000
00000000 00000000 00000000 00100000 00000001 00000000 00000000 00000000
00000000 00000000 00000000 01000000 00000001 00000000 00000000 00000000
00000000 10100000 00000000 01000000 00000001 00000000 00000000 00000000
11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
01110000 11111010 00000000 01000000 00000001 00000000 00000000 00000000
00101110 01100100 01111001 01101110 01100001 01101101 01101001 01100011
11110000 00000100 00000000 00100000 00000001 00000000 00000000 00000000
11110000 00000100 00000000 00100000 00000001 00000000 00000000 00000000
10110000 00000001 00000000 00000000 00000000 00000000 00000000 00000000
11110000 00000100 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
[...]
```

Source from <http://www.fas.harvard.edu/~csciel/distribution/lectures/11/compiled/hello.cc>.

9

# Languages

## Compiled

```
class Hello
```

```
{
```

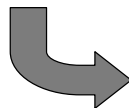
```
    public static void main(String [] argv)
```

```
    {
```

```
        System.out.println("Hello world!");
```

```
    }
```

```
}
```



```
11001010 11111110 10111010 10111110 00000000 00000011 00000000 00101101
00000000 00100000 00001000 00000000 00010100 00000111 00000000 00010011
00000111 00000000 00011010 00000111 00000000 00011011 00000111 00000000
00011100 00001010 00000000 00000100 00000000 00001001 00001001 00000000
00000101 00000000 00001010 00001010 00000000 00000011 00000000 00001011
00001100 00000000 00001111 00000000 00001100 00001100 00000000 00011110
00000000 00010110 00001100 00000000 00011111 00000000 00001101 00000001
00000000 00000011 00101000 00101001 01010110 00000001 00000000 00010101
00101000 01001100 01101010 01100001 01110110 01100001 00101111 01101100
01100001 01101110 01100111 00101111 01010011 01110100 01110010 01101001
01101110 01100111 00111011 00101001 01010110 00000001 00000000 00010110
00101000 01011011 01001100 01101010 01100001 01110110 01100001 00101111
01101100 01100001 01101110 01100111 00101111 01010011 01110100 01110010
01101001 01101110 01100111 00111011 00101001 01010110 00000001 00000000
00000110 00111100 01101001 01101110 01101001 01110100 00111110 00000001
00000000 00000100 01000011 01101111 01100100 01100101 00000001 00000000
00001101 01000011 01101111 01101110 01110011 01110100 01100001 01101110
01110100 01010110 01100001 01101100 01110101 01100101 00000001 00000000
00001010 01000101 01111000 01100011 01100101 01110000 01110100 01101001
01101111 01101110 01110011 00000001 00000000 00000101 01001000 01100101
[...]
```

Source from <http://www.fas.harvard.edu/~csciel/distribution/lectures/11/compiled/Hello.java>.

10

# JavaScript

```
<!DOCTYPE html
  PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html>
  <head>
    <script language="JavaScript" type="text/javascript">
      <!--
        var name = prompt("Please enter your name.", "");
        document.write("<title>Welcome, " + name + "!</title>");
      //-->
    </script>
  </head>
  <body>
    <center>
      <h1>
        Welcome,
        <script language="JavaScript" type="text/javascript">
          <!--
            document.write(name + "!");
          //-->
        </script>
      </h1>
    </center>
  </body>
</html>
```

Source from <http://www.fas.harvard.edu/~csciel/distribution/lectures/11/interpreted/name.html>.

11

# JavaScript

```
<!DOCTYPE html
  PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "DTD/xhtml1-transitional.dtd">

<html>
  <head>
    <title>Colorful Backgrounds</title>
  </head>
  <body>
    <center>
      <form action="">
        <input type="button" value="R"
          onclick="javascript:document.bgColor='red'"/>
        <input type="button" value="G"
          onclick="javascript:document.bgColor='green'"/>
        <input type="button" value="B"
          onclick="javascript:document.bgColor='blue'"/>
      </form>
    </center>
  </body>
</html>
```

Source from <http://www.fas.harvard.edu/~csciel/distribution/lectures/11/interpreted/colors.html>.

12

# JavaScript

```
<!DOCTYPE html
  PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "DTD/xhtml1-transitional.dtd">

<html>
  <head>
    <title>Blastoff</title>
  </head>
  <body>
    <center>
      <script language="JavaScript" type="text/javascript">
        <!--
          var i;
          for (i = 10; i >= 0; i = i - 1)
          {
            document.write(i + "...<br/>");
          }
          document.write("Blastoff!");
        //-->
      </script>
    </center>
  </body>
</html>
```

Source from <http://www.fas.harvard.edu/~csciel/distribution/lectures/11/interpreted/blastoff.html>.

13

# JavaScript

```
<!DOCTYPE html
  PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "DTD/xhtml1-transitional.dtd">

<html>
  <head>
    <title>My First Clock</title>
    <script language="JavaScript" type="text/javascript">
      <!--
        function startClock()
        {
          var now      = new Date();
          var hours    = now.getHours();
          var minutes  = now.getMinutes();
          var seconds  = now.getSeconds();

          document.clockform.hours.value  = "" + hours;
          document.clockform.minutes.value = "" + minutes;
          document.clockform.seconds.value = "" + seconds;

          timer = setTimeout("startClock()", 1000);
        }
      //-->
    </script>
  </head>
  [...]
</html>
```

Source from <http://www.fas.harvard.edu/~csciel/distribution/lectures/11/interpreted/clock1.html>.

14

# JavaScript

```
[...]
<body onload="startClock()">
  <center>
    <h1>My First Clock</h1>
    <br/>
    <form action="" name="clockform">
      <table>
        <tr>
          <td><input type="text" size="2" name="hours"/></td>
          <td><input type="text" size="2" name="minutes"/></td>
          <td><input type="text" size="2" name="seconds"/></td>
        </tr>
      </table>
    </form>
  </center>
</body>
</html>
```

Source from <http://www.fas.harvard.edu/~csciel/distribution/lectures/11/interpreted/clock1.html>.

15

# JavaScript

```
<!DOCTYPE html
  PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "DTD/xhtml11-transitional.dtd">

<html>
  <head>
    <title>My Second Clock</title>
    <script language="JavaScript">
      <!--
        function startClock()
        {
          var now      = new Date();
          var hours    = now.getHours();
          var minutes  = now.getMinutes();
          var seconds   = now.getSeconds();

          if (hours > 12) {
            document.clockform.hours.value = "" + hours - 12;
          }
          else {
            document.clockform.hours.value = "" + hours;
          }

          if (minutes < 10) {
            document.clockform.minutes.value = "0" + minutes;
          }
          else {
            document.clockform.minutes.value = "" + minutes;
          }
        }
      //--
    </script>
  </head>
  <body>
    <div id="clock">
      <div id="hours"><input type="text" value="" size="2" /></div>
      <div id="minutes"><input type="text" value="" size="2" /></div>
      <div id="seconds"><input type="text" value="" size="2" /></div>
    </div>
  </body>
</html>

[...]
```

Source from <http://www.fas.harvard.edu/~csciel/distribution/lectures/11/interpreted/clock2.html>.

16

# JavaScript

```
[...]
if (seconds < 10) {
    document.clockform.seconds.value = "0" + seconds;
}
else {
    document.clockform.seconds.value = "" + seconds;
}

if (hours >= 12) {
    document.clockform.ampm.value = "PM";
}
else {
    document.clockform.ampm.value = "AM";
}

timer = setTimeout("startClock()", 1000);
}
//-->
</script>
</head>
[...]
```

Source from <http://www.fas.harvard.edu/~csciel/distribution/lectures/11/interpreted/clock2.html>.

17

# JavaScript

```
[...]
<body onload="startClock()">
  <center>
    <h1>My Second Clock</h1>
    <br/>
    <form action="" name="clockform">
      <table>
        <tr>
          <td><input type="text" size="2" name="hours"/></td>
          <td><input type="text" size="2" name="minutes"/></td>
          <td><input type="text" size="2" name="seconds"/></td>
          <td><input type="text" size="2" name="ampm"/></td>
        </tr>
      </table>
    </form>
  </center>
</body>
</html>
```

Source from <http://www.fas.harvard.edu/~csciel/distribution/lectures/11/interpreted/clock2.html>.

18

# Computer Science E-1

Understanding Computers and the Internet

## Lecture 11: Programming

Thursday, 8 December 2005

David J. Malan  
malan@post.harvard.edu