Programming 3

Guessing Game

Don't need to know for exam!

Need to know for exam!

```
var rand = Math.random() // gives a random number between 0 and 1
var num = Math.round(rand*100) // num is an integer between 0 and 100
var max_guesses = 10 // maximum number of guesses allowed
var num_guesses = 0
var win = false
while (num_guesses < max_guesses && !win)
{
    var guess = prompt("Pick a number between 0 and 100", "")
    guess = parseInt(guess)
    if (guess > num) { alert(guess +" is too high!") }
    else if (guess < num) { alert(guess+" is too low!") }
    else if (guess == num) { win = true }
    num_guesses = num_guesses+1
}
if (win) { document.writeln("You win! :)") }
else { document.writeln("You lose :(") }
```

String Manipulation

- This function returns the reverse of a string.

```
function revstr(s)
{
   var c
   var newstr = ""// !!!
   var i = 0
   while (i <= s.length -1)
   {
     c = s.charAt(i)
     newstr = c + newstr
     i = i + 1
   }
   return newstr
}
document.writeln(revstr("Hello World!"))
// displays !dlroW olleH
```

String Manipulation

- This function also returns the reverse of a string.

```
function revstr(s)
{
   var c
   var newstr = ""// !!!
   var i = s.length -1
   while (i >= 0)
   {
    c = s.charAt(i)
    newstr = newstr + c
    i = i -1
   }
   return newstr
}
document.writeln(revstr("Hello World!"))
// displays !dlroW olleH
```

Decimal to Binary

```
var number = prompt("What decimal number (> 0) would you
  like to convert to binary?","10")
number = parseInt(number)
var answer = ""
document.writeln("The decimal number " + number + " in
  binary is ")
while (number > 0)
{
  remainder = number % 2
  answer = remainder + answer
  if (remainder > 0)
  {
    number = number - 1
  } // so that when we divide by 2,
    // we get no fractions
  number = number / 2
}
document.writeln(answer)
```

Binary to Decimal

```
var binary = prompt("What binary number (> 0)
  would you like to convert to decimal?","10")
var answer = 0
var position = binary.length - 1
var power = 0
var digit
while (position >= 0)
{
  digit = binary.charAt(position)
  digit = parseInt(digit)
  answer = answer + digit*Math.pow(2, power)
  power = power + 1
  position = position - 1
}
document.writeln("The binary number " + binary + "
  in decimal is " + answer)
```

Any Commas in a String?

```
function commas(s)
{
  var i = 0
  while (i < s.length)
  {
   var c = s.charAt(i)
   if (c == ",") {
     return true
   }
   i = i + 1
  }
  return false
}
document.writeln(commas("Hello, World!"))
// displays true
document.writeln(commas("Hey Marc!"))
// displays false
```

Tricky Conditionals

- Ask the user for a number between 1 and 5.
- Assume the user always enters a number; however, it can be any number. If the number entered is not between 0 and 5, “Invalid Input” should be written to the screen.
- Otherwise, if this number is even (recall the modulus operator %) and if it is greater than 2, then “Four” should be written to the screen.
- Otherwise, if the number is greater than 2, “Five or Three” should be written to the screen.
- Finally, in all other cases, “One or Two” should be displayed to the screen.

Tricky Conditionals

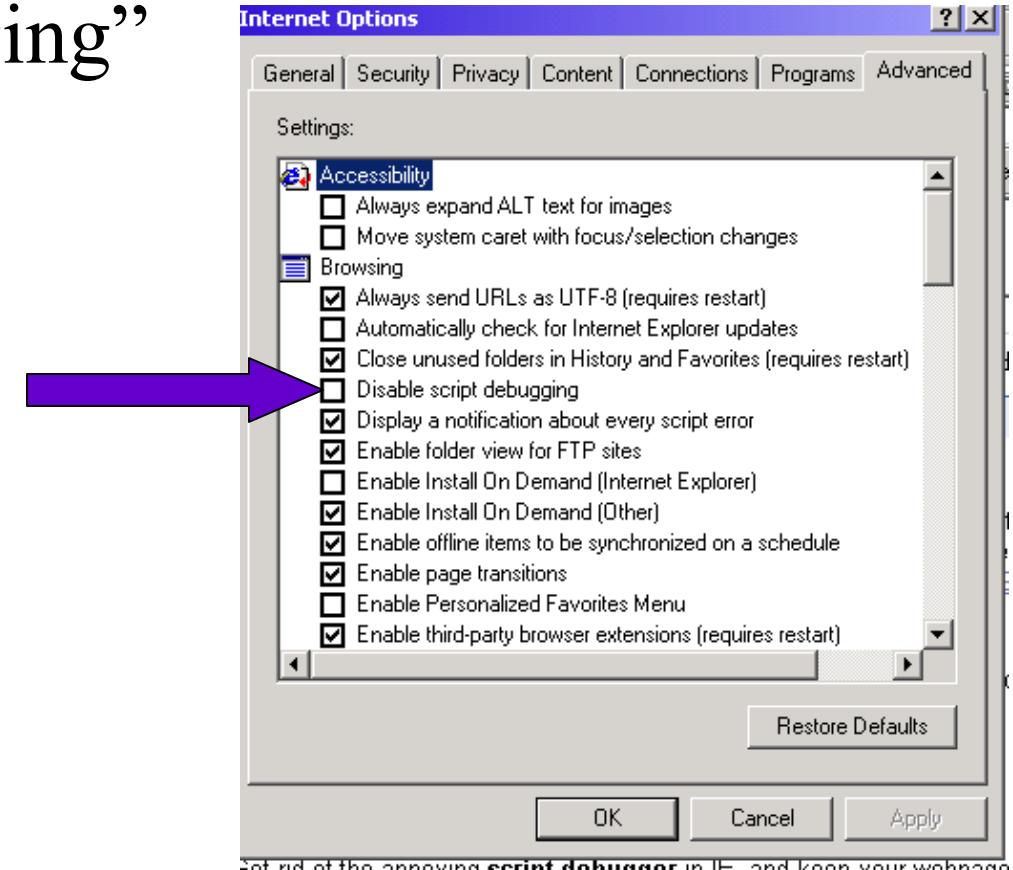
```
var num = prompt("Enter a number between 0 and 5", "")
num = parseInt(num)

if(num < 1 || num > 5)
{
  document.writeln("Invalid Input")
} else if(num %2 == 0 && num > 2)
{
  document.writeln("Four: "+ num)
} else if(num > 2)
{
  document.writeln("Five or Three "+ num)
} else
{
  document.writeln("One or Two "+ num)
}
```

Tips for writing your code

- If you copy and paste code, make sure that all the quotations are indeed “”.
- Debugging ideas:
  - Use document.writeln() statements to write the value of your variables to make sure that they store the correct value
  - Comment out sections of code that you don’t want to be executed
  - Build your code *incrementally*

Enable JavaScript Debugging in Internet Explorer

- Tools → Internet Options → Advanced and then *uncheck* the box that says "Disable script debugging"



Algorithm

- Origins of the word:
  - 9th century Muslim mathematician Abu Abdullah Muhammad ibn Musa al-Khwarizmi whose works introduced Arabic numerals and algebraic concepts.
  - The word *algorism* originally referred only to the rules of performing arithmetic using Arabic numerals
  - Evolved via European Latin translation of al-Khwarizmi's name into *algorithm* by the 18th century.
- The word evolved to include *all definite procedures for solving problems or performing tasks*.

Algorithm Design

- An **algorithm** is a finite set of well-defined instructions for accomplishing some task
- May be described
  - Abstractly using human language (pseudocode) describing the steps for carrying out some procedure using a computer
  - Using a programming language of your choice
  - By providing a set of machine instructions to be executed

Algorithm Design

- Pseudo-code is a programming language independent description of the sequence of steps necessary to solve a problem
- Algorithms that are written in pseudo-code may be then translated into a particular programming language to make a computer program
- A programmer may come up with his/her own algorithm, or (s)he may *implement* an existing algorithm

Algorithm Design

- Given a problem (e.g., find the largest number in a list of numbers, arrange the given numbers in sequence, convert from binary to decimal, etc.) seek the sequence of steps that will solve the problem
- Issues to be concerned with: efficiency
- Want to come up with a solution that uses computational resources (CPU cycles, storage) as efficiently as possible

Algorithm Design

- Coming weeks:
  - No more programming
  - Giving English-language (pseudocode) solutions to problems
  - Study examples of (relatively simple) problems in graph theory, searching, text encoding, computational geometry, …