Close look at Rock, Paper, Scissors

Rock, Paper, Scissors played face up hard-coded values: goal = to understand the logic

* userGuess = Rock
* computerGuess = Paper
* wins = 0
* losses = 0
* ties = 0 (here’s a difference between this and guessing game)
* gameNumber = 1 of 3

pseudocode

Click Button to start game – code snippet from W3Schools that we’ll hang execution off of

<https://www.w3schools.com/jsref/tryit.asp?filename=tryjsref_onclick>

gameNumber 1

if userGuess = computerGuess

then ties = ties + 1

if userGuess = Rock AND computerGuess = Scissors

then wins = wins + 1

if userGuess = Paper AND computerGuess = Rock

then wins = wins + 1

if userGuess = Scissors AND computerGuess = Paper

then wins = wins + 1

else losses = losses + 1

Repeat for 3 games, so gameNumber (1,2,3) – by convention let’s use a do while structure

Quick side note on if statement

* The what’s in parentheses after if, has to evaluate to either true or false

With AND, OR we get into Truth Tables

Let’s start with comparing 2 things for AND

* Let’s talk about parts A, B in terms of TRUE & FALSE
* We’ll sketch out all the possible combinations (or permutations?).
* AND is about BOTH being TRUE

|  |  |  |
| --- | --- | --- |
| The first thing we’ll call A | The second thing we’ll call B | A AND B |
| A = TRUE | B = TRUE | A && B = TRUE |
| A = TRUE | B = FALSE | A && B = FALSE |
| A = FALSE | B = TRUE | A && B = FALSE |
| A = FALSE | B = FALSE | A && B = FALSE |

OR is about one or the other or both

|  |  |  |
| --- | --- | --- |
| The first thing we’ll call A | The second thing we’ll call B | A OR B |
| A = TRUE | B = TRUE | A || B = TRUE |
| A = TRUE | B = FALSE | A || B = TRUE |
| A = FALSE | B = TRUE | A || B = TRUE |
| A = FALSE | B = FALSE | A || B = FALSE |

OR when we have 3 inputs, a few different ways to do this

Let’s take the above table for A OR B and OR it with C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| The first thing we’ll call A | The second thing we’ll call B | A OR B | C | A OR B OR C |
| A = TRUE | B = TRUE | A || B = TRUE | C = TRUE | A || B || C = TRUE |
| A = TRUE | B = FALSE | A || B = TRUE | C = TRUE | A || B || C = TRUE |
| A = FALSE | B = TRUE | A || B = TRUE | C = FALSE | A || B || C = TRUE |
| A = FALSE | B = FALSE | A || B = FALSE | C = FALSE | A || B || C = FALSE |

Why all the words?

A || B || C in our code is shown below, and our code block runs provided we feed it one of those inputs

if (userGuess == "r" || userGuess == "p" || userGuess == "s")

The AND Condition

If this evaluates to TRUE, which would require that \_\_\_\_\_\_\_\_\_\_\_\_

if(userGuess == "r" && computerGuess == "s")

then wins = wins + 1

That’s the main logic

Variables involved in the game

// Creates an array that lists out all of the options (Rock, Paper, or Scissors).

var computerChoices = ["r", "p", "s"];

// Grab the span ID's below and assign them to variables

var userText = document.getElementById("userGuess");

var compText = document.getElementById("compText");

var winText = document.getElementById("winText");

var lossText = document.getElementById("lossText");

var tieText = document.getElementById("tieText");

// COUNTERS

var userWins = 0;

var userLosses = 0;

var userTies = 0;

Trigger for the game logic to execute (Ignore debugger)

document.onkeyup = function (event) {

// To figure out WTF is going on with these variables...

debugger;

// Determines which key was pressed.

var userGuess = event.key;

// Randomly chooses a choice from the options array. This is the Computer's guess.

var computerGuess = computerChoices[Math.floor(Math.random() \* computerChoices.length)];