Curriculum Map

# 1.0 Welcome

## Essential Questions

* What are the roles of Erik and Richard?
* What is computer science?
* What is an algorithm?
* What is a data structure and what it is used for?
* How can we compare algorithms?

## Skills

* Be able to explain what computer science is
* Be able to explain what an algorithm is
* Ability to distinguish algorithms from non-algorithms

## Activities

* Find algorithms and data structures in grocery shopping scenario

# 1.1 Web page basics

## Essential Questions

* What makes up a web page?
* How does a browser retrieve a web page?
* What is a URL?
* What is a Resource?
* What is HTML?
* What is an HTML element? What is an HTML attribute?
* What is a HTML start tag and end tag?
* What does containment mean in HTML?
* What does it mean for HTML elements to automatically flow and why is that important?
* Why is HTML versioned?

## Skills

* Creating simple pages with a basic elements.
* Ability to use new tags due to understanding of the grammar.
* Using an editor to write HTML.
* Organizing and naming related files in a folder using good naming.
* Connecting additional files to the main HTML file using links.
* Mechanics of handing links in files.

## Activities

* Exercises

# 1.2 Functions, variables, and debugging

## Essential Questions

* What is the relationship between an algorithm and a function?
* What is a value?
* What is the relationship between variables and values?
* What is a global variable?
* What is a local variable?
* What is a call stack?
* What is a statement?
* What is an expression?
* What is pseudocode? (**INFORMATIONAL**)
* What is a data structure diagram?
* What is a function?
* What does calling a function do?
* What are the inputs and outputs to a function?
* How does executing a function affect data structures?
* What is a debugger for?
* Is a function a value? **(INTRODUCE THIS IN A LATER SECTION)**
* What is an event? (**MORE COVERAGE** **LATER**)
* What is a function call?
* How can I reuse an algorithm?
* How can I break down an algorithm into parts?
* Is a function the only way to create an algorithm?

## Skills

* Using a script element to connect Javascript to a web page
* Being able to recognize syntax errors
* Calling a function when a web page loads
* Using a debugger to find syntax errors
* Using a debugger to find runtime errors
* Using a debugger to verify results
* Writing functions with parameters and return values
* Breaking functions into parts
* Naming functions and variables
* Using local variables instead of global variables
* Using parameters instead of global variables

# 1.3 Objects

## Activity

* Step: Example via students
* Do: Line slope
* Step: Change students
* Do: Move point, then line slope

## Essential Questions

* What is an object and what is it used for?
* When do we use an object instead of separate parameters/variables?
* How are properties of an object accessed and updated?
* What is an object reference?

## Skills

* Using properties of objects in an expression.
* Using expressions when defining properties.
* Accessing properties of an object.
* Accessing nested properties (**LATER**)
* Working with object references.
* Understanding an object diagram.
* Creating object with or without properties.
* Updating the value of a property.

# 1.4 Strings and text output

## Activity

* Learning about character codes (ASCII, UNICODE)
* Learning about document objects
* Fill out a page template with student counts
* Adding styling (CSS)
* Adding HTML to a page (will require combining strings)
* Optional: create a table of values

## Essential Questions

* What is a string value?
* What is a character?
* What is ASCII and UNICODE?
* How are strings combined?
* What is the length of a string? **(LATER, WITH LOOPS)**
* What is escaping?
* How do we turn a number into a string?
* What are cascading style sheets?

## Skills

* How to link a stylesheet to the HTML
* Using the following properties of HTML elements
  + innerHTML
  + textContent
  + className
* Using text-related styling of CSS
  + Font family
  + Font size
  + Color and Background Color (named, with table)
* Combining strings.
* Using escape characters.
* Converting numbers to strings.
* Converting character codes to strings.
* Obtaining the object for an element on a web page.

# 1.5 Conditions and graphical output

## Activity

* Exploration of different types of graphics on a canvas
  + Mozilla developer network or similar
  + Free form activity
* Bouncing ball
  + Reflect when it nears the wall
  + Add second dimension
* Optional:
  + Add a second ball
  + Make them reflect bounce into each other
  + Figure out how to make the ball exactly touch the wall
  + Gravity

## Essential Questions

* What is a Boolean value?
* What are the operators in the Boolean algebra?
* What is an “if” statement for?
* What are the different forms for an “if” statement?
* What is a timer?
* How does animation work?
* What are milliseconds?
* How does a function do graphical output?
* What are different types of graphics output?

## Skills

* Translating logic statements in English into Boolean expressions.
* Using the different forms of if/else statements to control program flow.
* Creating a recurring timer.
* Converting between milliseconds and seconds
* Debugging with timers
* Working with simple graphics primitives to do output.
* Understanding two dimensional coordinate systems.

# 1.6 Iteration and Arrays

## Activities

* Informational instruction using soccer players
* Hands on instruction using bubbles animation

## Essential Questions

* What is iteration and what is it used for?
* What is a loop?
* What is an infinite loop?
* What is an array and what is it used for?
* How are elements of an array used and changed?
* What is the length of an array?
* What is an array index used to refer to a value?

## Skills

* Using loops to count.
* Using loops to continue until a condition.
* Using loops to step through arrays.
* Using nested loops. (**LATER**)
* Using break.
* Inserting items in an array (using function).
* Removing items from an array (using function).
* Appending item to an array (using function).

# 1.7 Input

## Activities

* Introduction to events with an example
  + Download sample which reports events
* Adding input to angry birds skeleton:
  + Mouse input
  + Key input: (up, down, left, right, space)
  + Form input: angle & velocity, launch button, validation
  + Reset button (to reset game)
  + Optional: touch events
  + Optional: window resize

## Essential Questions

* What are input element and what are they used for?
* How is input from the mouse, keyboard and touch modeled?

## Skills

* Creating pages using input elements.
* Writing functions that handle input.