

# Introduction to Programming

Class 4, 13 January 2017

Jack Phillips <[jack\\_phillips@asl.org](mailto:jack_phillips@asl.org)>

\$ sync

## Goals

**Goal 1:** You will know how to work with numbers.

**Goal 2:** You will know how to break a problem down into series of simpler problems.

## Vocabulary

Primitive Type  
Operator  
Number  
Evaluates  
Decomposition

## Code

```
parseInt()  
parseFloat()  
+, -, *, /  
Math.pow()  
Math.round()  
Math.ceil()  
Math.floor()  
.toString()  
.toFixed()
```

# Getting to know your partner

- **Share name and grade**
- **If you had to eat a worm, how would you cook it?**

## Review

What is an **ALGORITHM**?

## Review

What is an **ALGORITHM**?

A **series** of **finite** steps that will **solve a problem**.

## **Time to Grade**

1. Open Task04-Madlib
2. Swap computer with partner
3. Use rubric to grade partner's program

**Goal 1:** You will know how to work with numbers.

## Primitive Type

Definition: The simplest kind of data JavaScript can work with.

String

"Hello"

"Jack"

"5"

Number

-15

3.954

5



**Goal 1:** You will know how to work with numbers.

## Operator

Definition: An action you can use with a specific primitive.

### String

"Hello" + "Jack"

*evaluates to* "HelloJack"

"Hello".length

*evaluates to* 5

### Number

-15 + 4

*evaluates to* -11

Math.pow(5, 3)

*evaluates to* 125

**Goal 1:** You will know how to work with numbers.

## Operator

Definition: An action you can use with a specific primitive.

String

"5" + "2"

*evaluates to* "52"

Number

5 + 2

*evaluates to* 7

**Goal 1:** You will know how to work with numbers.

## Type Conversion

Definition: Changing one primitive type to another.

String -> Number

`parseInt("-2")`

*evaluates to* `-2`

Number -> String

`" " + -2`

*evaluates to* `"-2"`

`parseFloat("7.8")`

*evaluates to* `7.8`

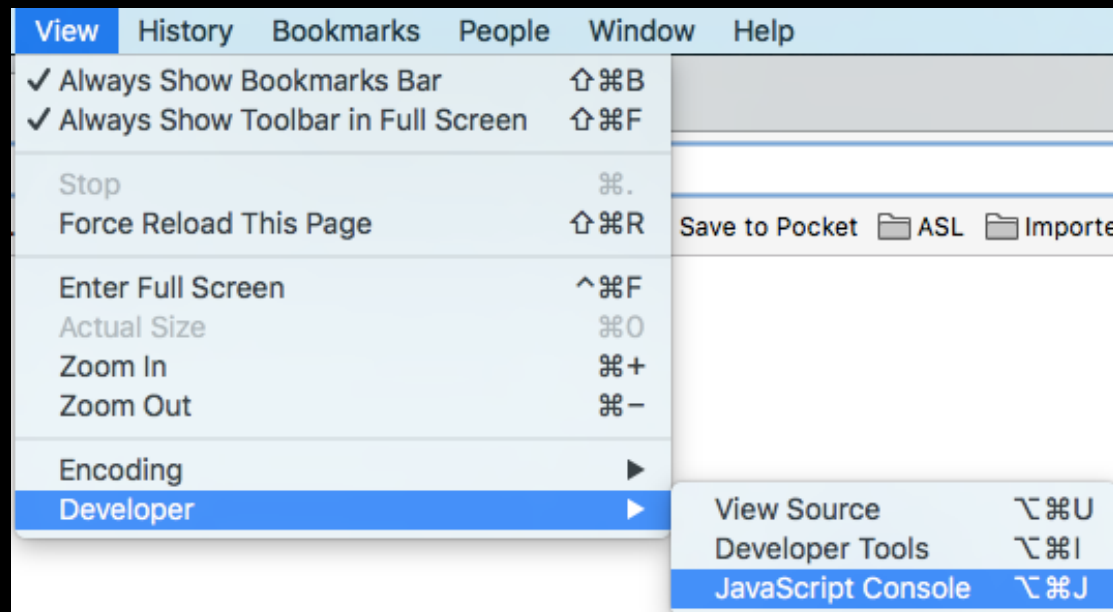
`7.8.toString()`

*evaluates to* `"7.8"`

**Goal 1:** You will know how to work with numbers.

## Let's do some math!

1. Open up the JavaScript Console.
2. Figure out the JavaScript code for each mathematical expression.



**Goal 1:** You will know how to work with numbers.

## Math Operations

Addition	+
Subtraction	-
Multiplication	*
Division	/
Exponentiation	<code>Math.pow(a, b)</code>
Round	<code>Math.round(a)</code>
Round Up	<code>Math.ceil(a)</code>
Round Down	<code>Math.floor(a)</code>
Round to n places	<code>a.toFixed(n)</code>

**Goal 1:** You will know how to work with numbers.

## Let's do some math!

<p><b>1 a.</b> <math>0.63 + 2^2</math></p> <p><code>0.63 + Math.pow(2, 2)</code></p>	<p><b>1 b.</b> <math>2 - 5.09 \cdot (1 + 3)^4</math></p> <p><code>2 - 5.09 * Math.pow((1 + 3), 4)</code></p>
<p><b>2 a.</b> <math>4.58 + 7 \cdot 2^3</math></p> <p><code>4.58 + 7 * Math.pow(2, 3)</code></p>	<p><b>2 b.</b> <math>\frac{4}{1.12 \cdot (-2)}</math></p> <p><code>4 / (1.12 * -2)</code></p>
<p><b>3 a.</b> <math>\frac{-2 + 7.63}{(3 + 4)^2}</math></p> <p><code>(-2 + 7.63) / Math.pow((3+4), 2)</code></p>	<p><b>3 b.</b> <math>(-2 + 6) \cdot (-6 + 7 + 9)</math></p> <p><code>(-2 + 6) * (-6 + 7 + 9)</code></p>
<p><b>4 a.</b> <math>-7.34 - 9.88 \cdot (-6.17) - 3</math></p> <p><code>-7.34 - 9.88 * -6.17 - 3</code></p>	<p><b>4 b.</b> <math>7.08 + 8 \cdot 0.68 - 2^5</math></p> <p><code>7.08 + 8 * 0.68 - Math.pow(2, 5)</code></p>
<p><b>5 a.</b> <math>-0.11 - 1 \cdot 1^2</math></p> <p><code>-0.11 - 1 * Math.pow(1, 2)</code></p>	<p><b>5 b.</b> <math>-9.15 - 10 \cdot 8.85 - (-4 + 1.24)</math></p> <p><code>-9.15 - 10 * 8.85 - (-4 + 1.24)</code></p>

**Goal 2:** You will know how to break a problem down into series of simpler problems.

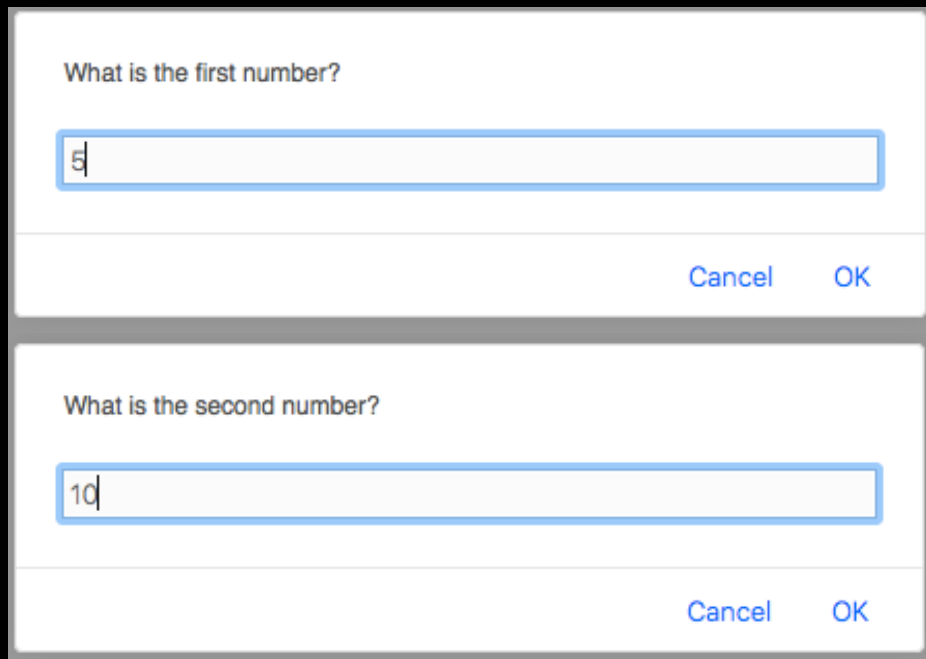
## Decomposition

Definition: Break a problem into smaller, easier problems.

**Goal 2:** You will know how to break a problem down into series of simpler problems.

## Task05-SimpleMath

Create a program called Task05-SimpleMath.html that prompts for two integers. Print the sum, difference, product, and quotient of those numbers as shown in the example output.



The image shows two separate input prompts. The first prompt asks 'What is the first number?' and has a text input field containing the number '5'. Below the input field are 'Cancel' and 'OK' buttons. The second prompt asks 'What is the second number?' and has a text input field containing the number '10'. Below this input field are also 'Cancel' and 'OK' buttons.

```
5 + 10 = 15  
5 - 10 = -5  
5 * 10 = 50  
5 / 10 = 0.5
```

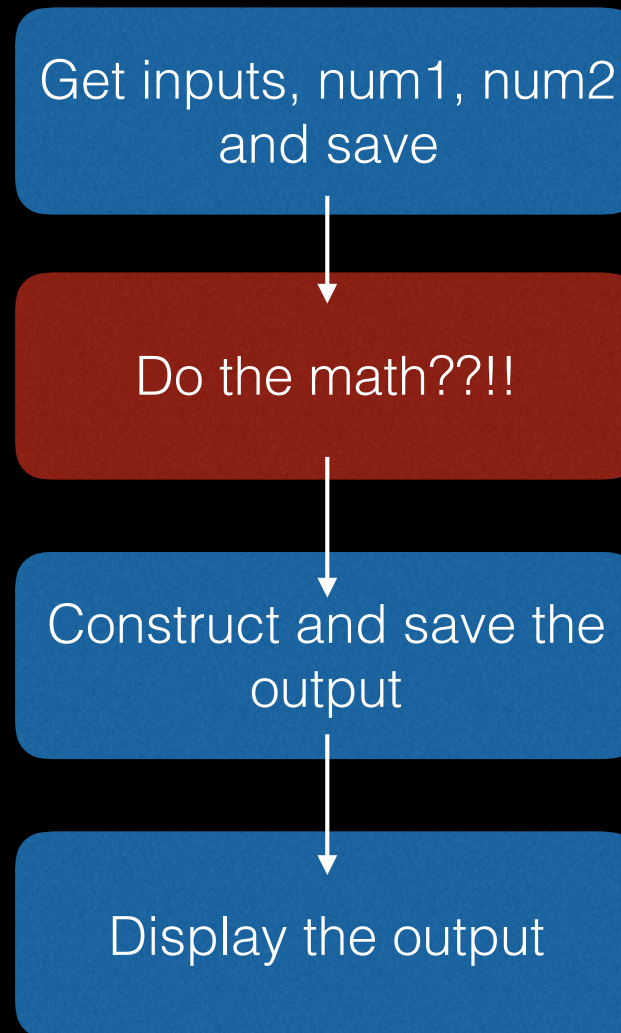
With your partner, write the algorithm/flowchart in English



**Goal 2:** You will know how to break a problem down into series of simpler problems.

## Flowchart

Brainstorm  
smaller, simpler  
problems.



**Goal 2:** You will know how to break a problem down into series of simpler problems.

## Simpler Problems

1. Get two numbers from the user and display the two numbers. 5, 10 -> "You entered 5 and 10"
2. Get two numbers from the user and display the sum of the two numbers. 5, 10 -> "15"
3. Get two numbers from the user and display the sum with the number sentence of the two numbers.  
5, 10 -> "5 + 10 = 15"

**Goal 2:** You will know how to break a problem down into series of simpler problems.

## Simpler Problems

1. Get two numbers from the user and display the two numbers. 5, 10 -> "You entered 5 and 10"

```
<script>
// Get the input as a string
var firstNumberString = prompt("What is the first number?"
);
var secondNumberString = prompt("What is the second
number?");

// Construct output
var output = "You entered " + firstNumber + " and " +
secondNumber;

// Generate output
alert(output);
</script>
```

**Goal 2:** You will know how to break a problem down into series of simpler problems.

## Simpler Problems

2. Get two numbers from the user and display the sum of the two numbers. 5, 10 -> "15"

```
<script>
// Get the input as a string
var firstNumberString = prompt("What is the first number?"
);
var secondNumberString = prompt("What is the second
number?");

// Convert the string responses into numbers
var firstNumber = parseInt(firstNumberString);
var secondNumber = parseInt(secondNumberString);

// Calculate the sum, difference, product, quotient
var sum = firstNumber + secondNumber;

// Generate output
alert(sum);
</script>
```

**Goal 2:** You will know how to break a problem down into series of simpler problems.

## Simpler Problems

3. Get two numbers from the user and display the sum with the number sentence of the two numbers.

5,  
10 -> "5 + 10 = 15"

```
// Get the input as a string
var firstNumberString = prompt("What is the first number?"
);
var secondNumberString = prompt("What is the second
number?");

// Convert the string responses into numbers
var firstNumber = parseInt(firstNumberString);
var secondNumber = parseInt(secondNumberString);

// Calculate the sum, difference, product, quotient
var sum = firstNumber + secondNumber;

// Construct output
var output = firstNumberString + " + " +
secondNumberString + " = " + sum;

// Generate output
alert(output);
```

**Goal 2:** You will know how to break a problem down into series of simpler problems.

## Simpler Problems

1. Get two numbers from the user and display the two numbers. 5, 10 -> "You entered 5 and 10"
2. Get two numbers from the user and display the sum of the two numbers. 5, 10 -> "15"
3. Get two numbers from the user and display the sum with the number sentence of the two numbers.  
5, 10 -> "5 + 10 = 15"
4. Go for it!

## Goals

**Goal 1:** You will know how to work with numbers.

**Goal 2:** You will know how to break a problem down into series of simpler problems.

## Vocabulary

Primitive Type  
Operator  
Number  
Evaluates  
Decomposition

## Code

```
parseInt()  
parseFloat()  
+, -, *, /  
Math.pow()  
Math.round()  
Math.ceil()  
Math.floor()  
.toString()  
.toFixed()
```