OSX calculator

Colors |

		Text
Buttons	operators : f79331 →	ffff (white)
	numbers: e0e0e0 →	0000 (black)
	AC : d6d6d6 →	(black)
	display : 746860 →	(white) [actually, it's shaded...]

id = "clear"

div id = "display"

class = "operator"

div id = "divide"

id = "multiply"

id = "subtract"

id = "add"

id = "calculate"

AC   CE   ✖

7   8   9   ÷
"seven" "eight" "nine"

4   5   6   ×
"four" "five" "six"

1   2   3   —
"one" "two" "three"

0   +
"zero"   "decimal"   =

• perform operations
• display result
• store result as initialValue

edge case:
only allow
one operator
in a row;
do not allow
operator as
first entry.
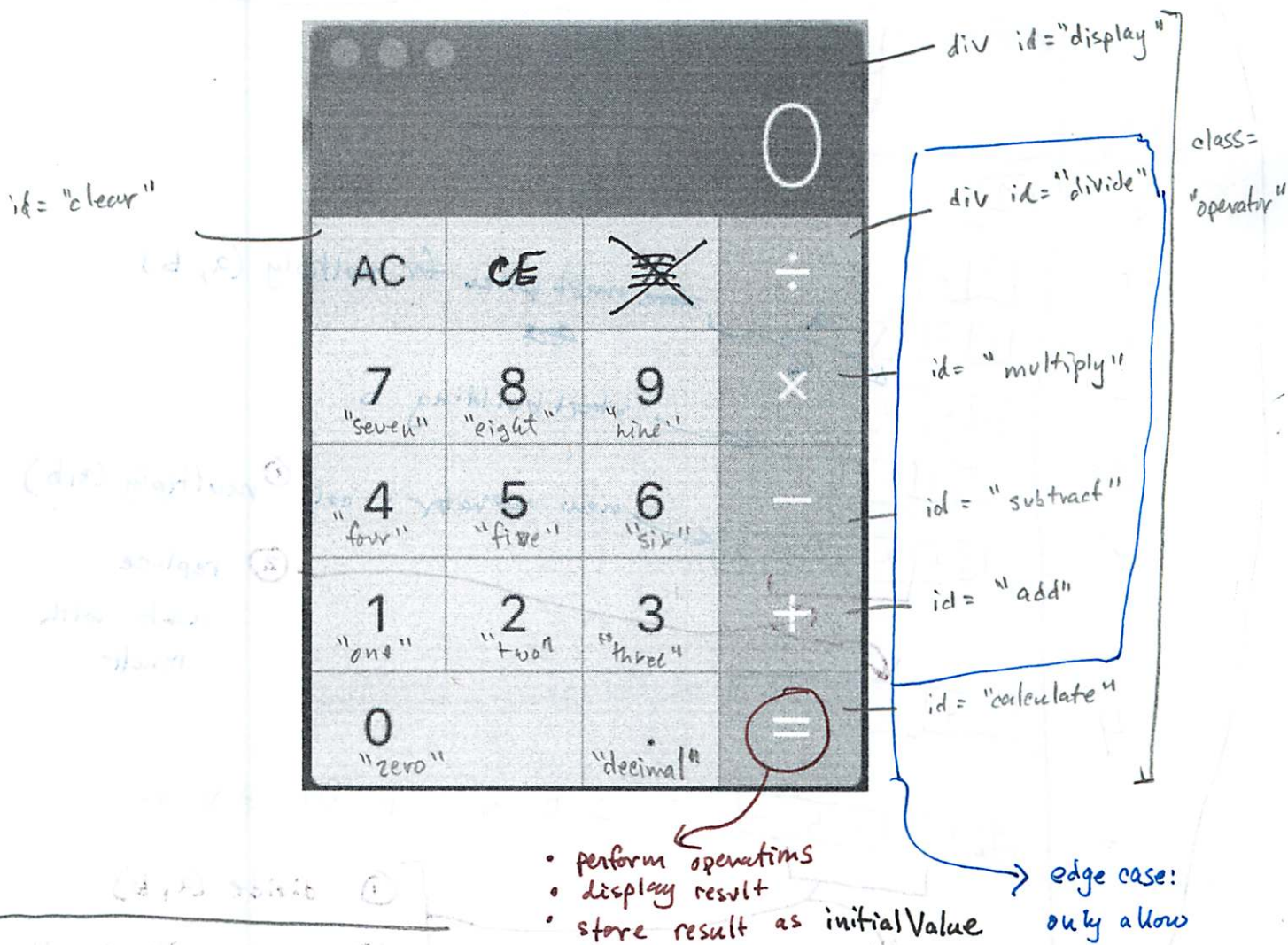
User stories:
• can add, subtract, multiply, and divide two numbers
• can clear the input field with a clear button
• can chain together mathematical operations until
  equal button is hit

Kem's BONUS: implement Order of Operations for
multiplication, division, addition, & subtraction

sample input:
$1 + 2 * 3 / 4 - 20 =$

a

b

↓

$1 + b - 20$

var initialValue = 0;
var inputArray = [];
function multiply (a, b)
function divide (a, b)
function calculateSum (array)
function clearEntry (array)
function allClear (array)
function displayResult ()
→ function numberBuilder

need **numberBuilder function**
- if multiple digits, join →
  [single digit elsewise]
inputArray • convert string to number
  return number

input	inputArray
1	`1`
+	`1` `+`
2	`1` `+` `2`
*	`1` `+` `2` `*`
3	`1` `+` `2` `*` `3`
/	`1` `+` `2` `*` `3` `/`
4	`1` `+` `6` `/` `4`
-	`1` `+` `6` `/` `4` `-`
20	`1` `+` `1.5` `-` `20`
=	`⊠`

store mark place for multiply (a, b)

a

start building b

new operator: call ① multiply (a, b)

② replace
a*b with
result

↓ Do similar for divide

① divide (a, b)
② replace with result

① on equal, calculate: **sum (array)**
② Display result, set inputValue
   to result