Golang

```
C:\Fall 19\CECS 424\Labs\Assn2>go run Calculator.go "1 + 3"
1 + 3 = 4
C:\Fall 19\CECS 424\Labs\Assn2>go run Calculator.go "2 - 6.4"
2 - 6.4 = -4.40
C:\Fall 19\CECS 424\Labs\Assn2>go run Calculator.go "5.6 * 7.8"
5.6 * 7.8 = 43.68
C:\Fall 19\CECS 424\Labs\Assn2>go run Calculator.go "8 / 4"
8 / 4 = 2
C:\Fall 19\CECS 424\Labs\Assn2>go run Calculator.go "8 / 4.0"
8 / 4.0 = 2.00
C:\Fall 19\CECS 424\Labs\Assn2>
```

```go
package main

import "fmt"
import "strings"
import "os"
import "strconv"
import "regexp"

func main() {

    args := os.Args[1]
    str := strings.Split(args, " ")
    var decimal = "\\d*(\\.\\d|\\d\\.)\\d*";

    match1, _ := regexp.MatchString(decimal, str[0])
    match2, _ := regexp.MatchString(decimal, str[2])

    if(match1 || match2) {
        //float arithmatic
        in1, err := strconv.ParseFloat(str[0], 64)
        in2, err := strconv.ParseFloat(str[2], 64)
        if err == nil {

            var ans = 0.1

            if str[1] == "+"{
                ans = in1 + in2
            } else if str[1] == "-" {
                ans = in1 - in2
            } else if str[1] == "*" {
                ans = in1 * in2
            } else {
                ans = in1 / in2
            }

            fmt.Printf(os.Args[1])
            fmt.Printf(" = %.2f", ans)
        }
    }else {
    //int arithmetic
        in1, err := strconv.Atoi(str[0])
        in2, err:= strconv.Atoi(str[2])
```

```
43          if err == nil {
44
45              var ans int
46
47              if str[1] == "+"{
48                  ans = in1 + in2
49              } else if str[1] == "-" {
50                  ans = in1 - in2
51              } else if str[1] == "*" {
52                  ans = in1 * in2
53              } else {
54                  ans = in1 / in2
55              }
56
57              fmt.Print(os.Args[1], " = ", ans)
58          }
59      }
60
61  }
```

Python

```python
import sys
import re

def main(argv):
  text = argv[1]
  args = text.split(" ")
  regExpr = "\\d*(\\.\\d|\\d\\.)\\d*"
  x = re.search(regExpr, args[0])
  y = re.search(regExpr, args[2])
  if x or y:
    int1 = float(args[0])
    int2 = float(args[2])
    if args[1] == "+":
      output = int1 + int2
    elif args[1] == "-":
      output = int1 - int2
    elif args[1] == "*":
      output = int1 * int2
    else:
      output = int1 / int2
    print(argv[1], "=", output)
  else:
    int1 = int(args[0])
    int2 = int(args[2])
    if args[1] == "+":
      output = int1 + int2
    elif args[1] == "-":
      output = int1 - int2
    elif args[1] == "*":
      output = int1 * int2
    else:
      output = int1 / int2
    print(argv[1], "=", "{:.0f}".format(output))

main(sys.argv)
```

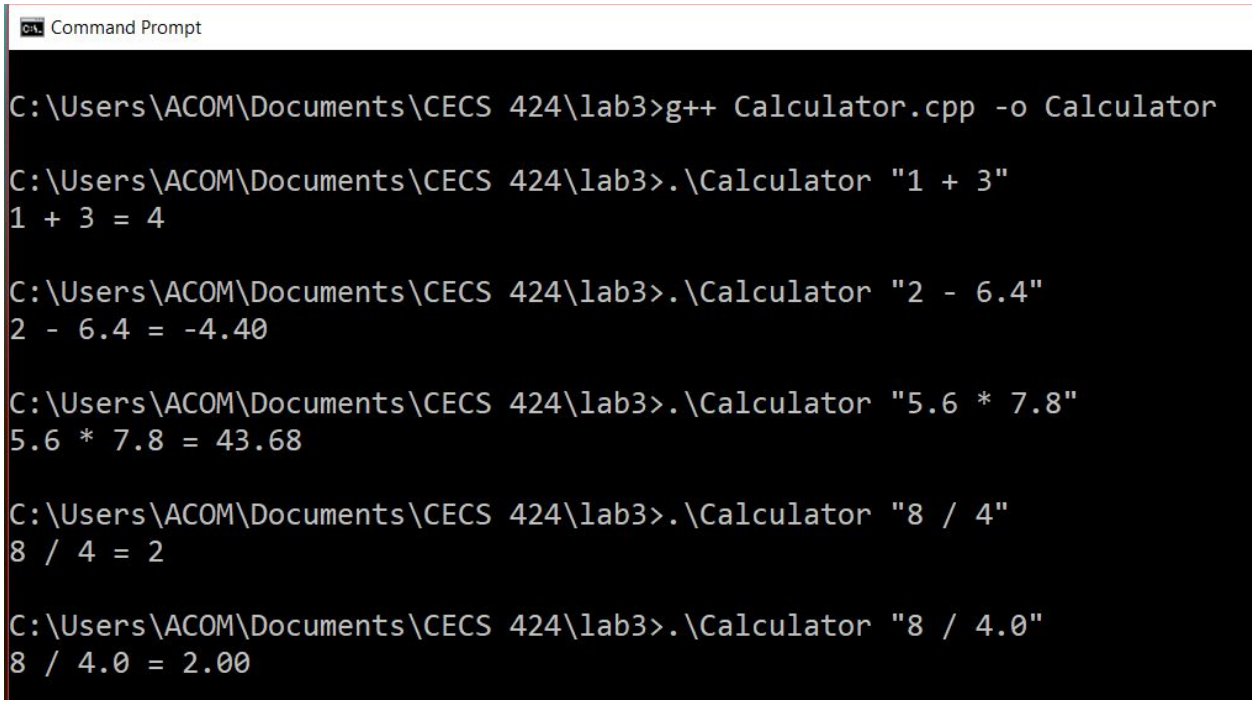
```
Microsoft Windows [Version 10.0.18362.295]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\scott\Documents\CECS 424>python Calculator.py "1 + 3"
1 + 3 = 4

C:\Users\scott\Documents\CECS 424>python Calculator.py "2 - 6.4"
2 - 6.4 = -4.4

C:\Users\scott\Documents\CECS 424>python Calculator.py "5.6 * 7.8"
5.6 * 7.8 = 43.68

C:\Users\scott\Documents\CECS 424>python Calculator.py "8 / 4"
8 / 4 = 2

C:\Users\scott\Documents\CECS 424>python Calculator.py "8 / 4.0"
8 / 4.0 = 2.0

C:\Users\scott\Documents\CECS 424>
```

Javascript

```
var args = process.argv[2].split(" ")
var regExpr = new RegExp('\\d*(\\.\\d|\\d\\.)\\d*');
var operation = args[1]
var output
if(regExpr.test(args[0]) || regExpr.test(args[2])){
  var int1 = parseFloat(args[0])
  var int2 = parseFloat(args[2])
}
else{
  var int1 = parseInt(args[0])
  var int2 = parseInt(args[2])
}
if(operation === '+'){
  output = int1 + int2
}
else if(operation === '-'){
  output = int1 - int2
}
else if(operation === '*'){
  output = int1 * int2
}
else{
  output = int1 / int2
}
output.toFixed(2)
var result = process.argv[2] + ' = ' + output
console.log(result)
```

```
Microsoft Windows [Version 10.0.18362.295]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\scott\Documents\CECS 424>node Calculator.js "1 + 3"
1 + 3 = 4

C:\Users\scott\Documents\CECS 424>node Calculator.js "2 - 6.4"
2 - 6.4 = -4.4

C:\Users\scott\Documents\CECS 424>node Calculator.js "5.6 * 7.8"
5.6 * 7.8 = 43.68

C:\Users\scott\Documents\CECS 424>node Calculator.js "8 / 4"
8 / 4 = 2

C:\Users\scott\Documents\CECS 424>node Calculator.js "8 / 4.0"
8 / 4.0 = 2

C:\Users\scott\Documents\CECS 424>
```

C++




```
#include<iostream>
#include<sstream>
#include<regex>
using namespace std;

int main(int argc, char** argv)
{
	string s(argv[1]);
	stringstream ss(s);
	string tokens[3];
	string token;
	for(int i = 0; i < 3; i++)
	{
		getline(ss, token, ' ');
		tokens[i] = token;
	}

	string op = tokens[1];
	regex decimal("\\d*(\\d*\\.|\\.\\d*)\\d*");
	if(regex_match(tokens[0], decimal) || regex_match(tokens[2], decimal))
	{
```

```
		double d1 = stod(tokens[0]);
		double d2 = stod(tokens[2]);
		if(op.compare("+") == 0)
		{
			cout<<s<< " = ";
			printf("%.2f", d1+d2);
			cout<<endl;
		}
		else if(op.compare("-") == 0)
		{
			cout<<s<< " = ";
			printf("%.2f", d1-d2);
			cout<<endl;
		}
		else if(op.compare("*") == 0)
		{
			cout<<s<< " = ";
			printf("%.2f", d1*d2);
			cout<<endl;
		}
		else
		{
			cout<<s<< " = ";
			printf("%.2f", d1/d2);
			cout<<endl;
		}

	}
	else
	{
		int n1 = stoi(tokens[0]);
		int n2 = stoi(tokens[2]);
		if(op.compare("+") == 0)
			cout<<s<< " = "<<n1 + n2<<endl;
		else if(op.compare("-") == 0)
			cout<<s<< " = "<<n1 - n2<<endl;
		else if(op.compare("*") == 0)
			cout<<s<< " = "<<n1 * n2<<endl;
		else
			cout<<s<< " = "<<n1 / n2<<endl;
	}

	return 0;
```

```
}
```