

## Closed Lab 0

## Math Operations and Print Statements

Description: For this lab, you will write a short Python program that prints the integers from 0 through 20, in order. Each of the 21 numbers should be printed on a separate line. That's it!

Hmmm, you know what, that doesn't have much flair. Let's make it just a bit more interesting. Instead of printing the numbers directly (as integer literals), let's generate each of them with an arithmetic expression. Each arithmetic expression must contain exactly four 4s and no other numbers. You may use parentheses, type casting, and the following operations:

Operation	Syntax
addition	+
subtraction	-
multiplication	*
division	/
square root	<code>math.sqrt()</code>
exponentiation	**
factorial	<code>math.factorial()</code>

Note the `import` statement near the top of the provided code file. It makes the `math` functions available. To type cast in Python, you provide the type followed by the value/variable to be cast. For example, to cast variable `num` to type `int`: `int(num)`.

Each line in your program will include a `print` statement that shows the fours and operations you use to achieve the desired value, followed by the arithmetic expression. For example, to print 0 you could write this line of code:

```
print('4 + 4 + 4 + 4 = {}'.format(4+4+4+4))
```

which results in this output:

$$4 + 4 + 4 + 4 = 16$$

In that statement, the first part is a string literal. The braces serve as a placeholder for the expression in the `format` call. Note that `format` is a method of the `String` class.

You could write the line of code but you won't because that is a completely uninspired way to achieve 16 using four 4s, a way that will not win you the undying admiration of your peers.

One last thing: Yes, it is possible to generate all integers in `[1..20]` in this way.