

---

## CMSC 201 Fall 2015

### Homework 2 – Order of Operations and Modulo

**Assignment:** Homework 2 – Order of Operations and Modulo

**Due Date:** Tuesday, September 15th, 2015 by 8:59:59 PM

**Value:** 4% of final grade

Homework 2 is designed to help you practice order of operations and modulo (or “mod”) in a Python environment. Remember to enable Python 3:

```
/usr/bin/scl enable python33 bash
```

As a reminder, mod is the remainder left over when one number is divided by another. For example,  $7 \% 3 = 1$ . The value of mod will always be from zero – where there is no remainder (e.g.,  $10 \% 5 = 0$ ), to 1 less than the mod number (e.g.,  $19 \% 5 = 4$ ).

### Instructions

**For the first 8 questions below**, there are three steps.

1. Using a comment in your python code, hypothesize what the correct answer is using order of operation.
2. Code the expression using a print command.
3. Comment on why your answer was correct or incorrect.

**For the last 2 questions below**, you need to add parentheses to the equation so that it evaluates to the target number given. Make sure you test that your solution works using Python.

Make sure that you **think about and hypothesize the answer before coding it** up and having Python calculate the answer. (You won’t have a Python interpreter available on the midterm, after all.)

**At the end, your hw2.py file must run without any errors.**

NOTE: Python may give you a decimal for your whole number (e.g.,  $12.0$  instead of  $12$  for Example 3 below). We’ll discuss the reason for this later.

### Example 1:

Evaluate the following expression:

`example1 = (10 + 5) * 5 - 3`

1. What do you think `example1` equals?
2. Code the expression
3. Were you correct? Why or why not?

```
# Example 1:
# Expected output: 72
example1 = (10 + 5) * 5 - 3
print("Example 1 evaluates to: ", example1)
# Actual output: 72
# Explanation: Parentheses first (15), then multiplication (75),
#              then subtraction (72)
```

### Example 2:

Evaluate the following expression:

`example2 = (5 + 5) * 5 - 3 % 3`

```
# Example 2:
# Expected output: 49
example2 = (5 + 5) * 5 - 3 % 3
print("Example 2 evaluates to: ", example2)
# Actual output: 50
# Explanation: Parentheses first (10), then multiplication (50)
#              and modulo (0), then subtraction (50)
#              My mistake was 3 % 3 = 0, and I thought it was 1.
```

### Example 3:

Add parentheses so that the equation evaluates to the target number:

`example3 = (8 * 3) / 6 * (1 + 2)`

`target3 = 12`

```
# Example 3:
# Given equation: 8 * 3 / 6 * 1 + 2
# Solved equation: (8 * 3) / 6 * (1 + 2)
# Target number: 12
example3 = (8 * 3) / 6 * (1 + 2)
print("Example 3 evaluates to: ", example3, " and should be ", 12)
```

## Questions

### Question 1

$$\text{num1} = (7 + 1) * 3$$

### Question 2

$$\text{num2} = (12 \% 5)$$

### Question 3

$$\text{num3} = (21 \% 49)$$

### Question 4

$$\text{num4} = (5 - 3) + (10 - 5) * (8 \% 2)$$

### Question 5

$$\text{num5} = 6.5 + 5 / 2 * (4 + 7)$$

### Question 6

$$\text{num6} = 9 / 3 + 18 - 4 * 4$$

### Question 7

$$\text{num7} = 8 \% 3 + 5 * 4$$

### Question 8

$$\text{num8} = 81.3 / 2.1 + ((51.5 \% 65.2) * 2 / 2.5)$$

### Question 9

Add parenthesis where needed

$$\text{num9} = 100 - 8 * 8 + 1 / 0.5$$

$$\text{target9} = -30$$

### Question 10

Add parenthesis where needed

$$\text{num10} = 84 / 10 + 11 - 4 * 4$$

$$\text{target10} = 0$$

## Submitting

Once your `hw2.py` file is complete, it is time to turn it in with the `submit` command.

**Don't forget to complete the header block** for `hw2.py`! If you need a reminder of what this should be, see the video in Part 1 of Lab 2.

You must be logged into your GL account, and you must be in the same directory as the `hw2.py` file. To double check this, you can type `ls`.

```
linux1[3]% ls
hw2.py
linux1[4]% █
```

To submit your `hw2.py` file, we use the `submit` command, where the class is `cs201`, and the assignment is `HW2`. Type in `submit cs201 HW2 hw2.py` and press enter.

```
linux1[4]% submit cs201 HW2 hw2.py
Submitting hw2.py...OK
linux1[5]% █
```

If you don't get a confirmation like the one above, check that you have not made any typos or errors in the command.

You can double-check that your homework was submitted by using the `submitls` command. Type in `submitls cs201 HW2` and hit enter. (You'll see your own username, of course.)

```
linux1[5]% submitls cs201 HW2
total 21
drwx-----  2 mneary1 rpc  2048 Aug 30 19:01 .
drwx----- 507 mneary1 rpc 18432 Aug 30 16:15 ..
-rw-r--r--   1 k38      rpc   167 Sep  6 19:01 hw2.py
linux1[6]% █
```

And you're done!