

CSCI 150 HW: function and loop reading practice

Due: Monday, September 24

To receive full credit, for each exercise you should do the following:

1. **Predict:** First, complete the exercise *without* using the Python interpreter. (You are welcome to refer to your notes or textbook, read Python documentation, look at examples from class, *etc.*; just don't actually run any code.) Trace the execution of the code in the exercise.
2. **Check:** Run the code. Does the actual output agree with what you wrote down in step 1?
3. **Evaluate:** If your answer to part 1 was different than the actual output, keep experimenting with it, consult the textbook or Python documentation, ask a friend or TA or professor, *etc.* until you can explain why the code works the way it does *and* what your misunderstanding(s) were in part 1. (You do not need to do anything for step 3 if the output agrees exactly with what you wrote in step 1.)

You will not be graded on how correct your answer is in part 1. However, you *will* be graded on the accuracy of your evaluation in step 3. Obviously, I will not be able to tell the difference if you simply run the code and paste the output for step 1; please do not do that! You will only be depriving yourself of a learning opportunity (not to mention that it is a violation of the academic integrity policy).

Turn in your answers and evaluations either electronically via the usual form, or on paper.

You should consider the code in each exercise separately from the other exercises.

1. Consider the functions defined below. Trace the execution when `main1()` is called.

```
def foo(a: int) -> int:
    b = 3*a + 2
    return b
    print("In foo")

def bar(x: int, y: int) -> int:
    return foo(x) + foo(y)

def main1():
    print("The value is " + str(bar(2,3)))

main1()
```

2. Consider the functions defined below. Trace the execution when `main2()` is called.

```
def f1():
    print("mushroom")

def f2():
    f1()
    print("badger")
    f1()

def f3(n: int):
    f2()
    if n > 5:
        print("snake")
        f1()
    else:
        print("snaaaaaake")

def main2():
    f3(2)
    f3(6)
```

`main2()`

3. Trace the execution when `main3` is called.

```
def main3():
    s: int = 0
    i: int = 0
    while i < 5:
        j: int = 0
        while j < i:
            s += j
            j += 1
        i += 1
    print(s)
```

`main3()`