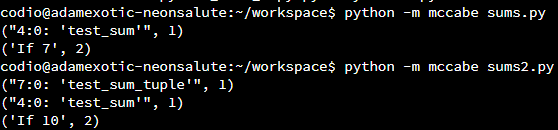
**Questions:**

Run *mccabe* on sums.py. What is the result?

Run *mccabe* on sums2.py. What is the result?

What are the contributors to the cyclomatic complexity in each piece of code?



In the first python code, the main contributor to cyclomatic complexity (CC) is if statement and it accounts for 2 units. The if statement is on the line 7 while the function test\_sum is on the 4th line and it’s complexity is 1 unit.

The second script has two functions each with CC of 1 unit and an if statement which has 2 units of cyclomatic complexity, same as the previous script.

This software metric effectively measures the independent paths in the code, to assess the complexity of a program.

**Appendices:**

**sums.py**

*# SOURCE OF CODE: https://realpython.com/python-testing/*

def test\_sum():

    assert sum([1, 2, 3]) == 6, "Should be 6"

if \_\_name\_\_ == "\_\_main\_\_":

    test\_sum()

    print("Everything passed")

**sums2.py**

*# SOURCE OF CODE: https://realpython.com/python-testing/*

def test\_sum():

    assert sum([1, 2, 3]) == 6, "Should be 6"

def test\_sum\_tuple():

    assert sum((1, 2, 2)) == 6, "Should be 6"

if \_\_name\_\_ == "\_\_main\_\_":

    test\_sum()

    test\_sum\_tuple()

    print("Everything passed")