Name:

ID:

Submit here: https://docs.google.com/forms/d/e/1FAIpQLSdNYDOm8ZBoK3Kw88Jkbl16cWULsWYsFRJo42nact1Gj_QtbQ/viewform

Python editor: https://repl.it/repls/IrresponsibleWrathfulAddresses

M5 Challenge 05D

1. Consider the following:

```
1  def collatz(n):
2    while n != 1:
3      yield(n)
4      n = n/2 if n%2 == 0 else 3*n + 1
5
6  _iter = collatz(20)
7  print(next(_iter))
8  print(next(_iter))
```

- How many times should I print **next(_iter)** to get the value 2?

4	7
6	11
9	5

- Let n = 91. How long is this collatz sequence? (Include n and 1 in the collatz sequence. For example, collatz(20) has a length of 8: 20, 10, 5, 16, 8, 4, 2, 1)

93	96
105	54
107	91
109	111

2. Consider the following:

```
"""

A = [a1 a2]
    [a3 a4]

B = [b1 b2]
    [b3 b4]

A x B = [(a1*b1 + a2*b3) (a1*b2 + a2*b4)]
        [(a3*b1 + a4*b3) (a3*b2 + a4*b4)]

"""


def multiply(A,B):
  return [
    [],
    []
  ]

```

- Lines 1-12 describe how to multiply two 2x2 matrices. Complete the multiply function so that it correctly multiplies two matrices A and B. Matrices will be represented as list with two lists inside: [[],[]]. Paste a screenshot of your code below:
- Complete the following table:

A	B	A x B
[[1,0],[0,1]]	[[3,1],[9,1]]	
[[100,-20.2],[200.1,-0.4]]	[[2.611,1.4429],[4.1,-1.93]]	
[[0.1245,0.1233],[0.155,0.13456]]	[[0.22,0.44],[0.22,11.1345]]	

3. Consider the following:

```

  def det(A):
    return A[0][0]*A[1][1] - A[0][1]*A[1][0]

  def inv(A):
    return [[1/det(A)*A[1][1],-1/det(A)*A[0][1]],
            [-1/det(A)*A[1][0],1/det(A)*A[0][0]]]

```

- Give A is a 2x2 matrix represented by a list of two lists. Complete the following table:

A	det(A)	inv(A)	multiply(A,inv(A))
[[1,3],[1,1]]			
[[3,2],[0.3,0.3]]			
[[-3,1.66],[0.51,2.52]]			