**Learning Journal Unit 3**

Part 1:

* The code of my program:

>>>

# the countdown function

def countdown(n):

if n <= 0:

print('Blastoff!')

else:

print(n)

countdown(n-1)

# the countup function

def countup(n):

if n >= 0:

print('Blastoff!')

else:

print(n)

countup(n+1)

num = int(input('Give number: '))

# ask is the num is pos or neg

if num >= 0:

countdown(num)

else:

countup(num)

* Respective output for the following inputs: a positive number, a negative number, and zero:

Positive number:

Give number: 4

4

3

2

1

Blastoff!

Negative number:

Give number: -6

-6

-5

-4

-3

-2

-1

Blastoff!

Zero:

Give number: 0

Blastoff!

* An explanation of your choice for what to call for input of zero:

The program calls the function countdown() because in the if condition I told the program if num is greater than or equal to 0 then do the countdown function.

Part 2:

Write your own unique Python program that has a **runtime error**:

* The code for my program:

>>>

password = '1234'

input\_pass = input('type your password: ')

# check if the password is correct

def ask\_password(x, password):

if password == x:

print('Correct password')

else:

print('Incorrect password')

ask\_password(x, password)

ask\_password(password, input\_pass)

* Output demonstrating the runtime error, including the error message:

Traceback (most recent call last):

File "/Users/yousefadel/Documents/Computer Science/Python/get to zero.py", line 38, in <module>

ask\_password(password, input\_pass)

File "/Users/yousefadel/Documents/Computer Science/Python/get to zero.py", line 36, in ask\_password

ask\_password(x, password)

File "/Users/yousefadel/Documents/Computer Science/Python/get to zero.py", line 36, in ask\_password

ask\_password(x, password)

File "/Users/yousefadel/Documents/Computer Science/Python/get to zero.py", line 36, in ask\_password

ask\_password(x, password)

[Previous line repeated 993 more times]

File "/Users/yousefadel/Documents/Computer Science/Python/get to zero.py", line 35, in ask\_password

print('Incorrect password')

RecursionError: maximum recursion depth exceeded while calling a Python object

* An explanation of the error message:

The program print RecursionError because the function has called it self for 1000 time.

* An explanation of how to fix the error:

I fix the error by putting the input function inside the ask\_password function so that the program asks the user to input the password again until he gets it right like the following:

>>>

password = '1234'

# check if the password is correct

def ask\_password(x):

password = input('type your password: ')

if password == x:

print('Correct password')

else:

print('Incorrect password')

ask\_password(x)

ask\_password(password)