CSC 2280

Introduction to Computer Science

Florida Southern College

**Lab 2**

**Friday, February 1, 2019**

The following exercises are intended to improve your understanding of computer science, and specifically Python programming, through guided practice. Follow the instructions carefully and make sure that requested input/output looks identical to the input/output of your code.

**Notes:**

* For this lab, you may use any valid Python IDE or text editor + shell
* Spend some time familiarizing yourself with the mathematical operations built-in to Python as well as range and for loops

1. What is the output of the following Python commands? Write your answer below each statement. If the code produces an error, explain why. *Try to predict what the output will be before running the code to see if it matches your intuition!*

print(10.0 / 4.0 + 3.5 \* 2)

9.5 Float

print(10 % 6 + 4 / 2)

6.0 Float

print(abs(5 - 21 // 4) \*\* 4)

0 Integer

print(sqrt(4.5 - 5.0) + 7 \* 3)

Error message, saying square root is not a function

print(4 \* 11 // 4 + 11 % 4)

14 Integer

print(5 \*\* 5)

3125 Integer

print(10 \*\* 5 + 10 \*\* 4 + 10 \*\* 3 + 10 \*\* 2 + 10 \*\* 1 + 10 \*\* 0)

11111 Integer

print(81 \*\* (1/2) - 64 \*\* (1/3) + 4096 \*\* (1/4) - 128 \*\* (1/7))

11.0 Float

1. For each of the previous expression in (1), what is the data type of the output? Write your answer next to or below each of the results.
2. In your own words, what does the function range do?

Lists out all the values from either 0 to the given value, or between two given values

1. To see the list of numbers generated by range, you can use the list function! Try out the following lines of code to see how range works.

list(range(0))

[]

list(range(1))

[0]

list(range(10))

[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

list(range(-10))

[]

list(range(5, 10))

[5, 6, 7, 8, 9]

list(range(-3, 3))

[-3, -2, -1, 0, 1, 2]

list(range(0, 100, 10))

[0, 10, 20, 30, 40, 50, 60, 70, 80, 90]

list(range(4, 13, 3))

[4, 7, 10]

list(range(0, 20, -1))

[]

x, y, z = 15, 5, -2

list(range(x, y, z)))

1. Convert the following mathematical expressions to Python code. You may assume that any relevant variables will have been assigned already.

c = math.sqrt(a \*\* 2 + b \*\* 2)

**import** math  
  
a = 4  
b = 8  
c = math.sqrt(a \*\* 2 + b \*\* 2)  
  
g = ((b \*\* 2) - (4 \* a \* c)) / 2 \* a  
  
x = (math.sqrt(g)) \* (-1 \*b)

I could not make the last line of code work for the life of me

m = (n \* (n -1)) / 2

V = (4 /3) \* math.pi \* (r \*\* 2)

E = m \* (c \*\* 2)

1. Write the Python code using range() that would generate the following sequences of numbers.

[1, 2, 3, 4, 5]

List(range(1, 6))

[5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90]

List(range(5, 95, 5)

[-17, 0, 17]

List(range(-17, 18, 17)

[10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0]

List(range(10, -1, -1))

[500, 350, 200, 50, -100, -250, -400]

List(range(500, -450, -150)

1. Show the output for the following Python code fragments.

for i in [1,3,5,7,9]:

print(i, ":", i\*\*3)

print(i)

1 : 1

3 : 27

5 : 125

7 : 343

9 : 729

9

x = 2

y = 10

for j in range(0, y, x):

print(j, end="")

print(x + y)

print("done")

012

212

412

612

812

done

ans = 0

for i in range(1, 11):

ans += ans + i\*i

print(i)

print(ans)

5

6

7

8

9

10

5998

1. Write a program to sum a series of numbers entered by the user. The program should first prompt the user for how many numbers are to be summed. The program should then prompt the user for each of the numbers in turn and print out a total sum after all the numbers have been entered.  
   *HINT: Use an input statement in the body of the loop!*

n = int(input(**"How many numbers would you like summed?: "**))  
total = 0  
  
**for** i **in** range(n):  
 y = int(input(**"What numbers?: "**))  
 total += y  
  
print(total)

1. Write a program that approximates the value of pi by summing the terms of this series:

The program should prompt the user for *n*, the number of terms to sum, and then output the sum of the first *n* terms of this series. After you print the final result, also print the error from the actual value of pi.

**def** main():  
 x = 1  
 f = 4  
 z = 0  
 r = int(input(**"How long would you like your pi?: "**))  
 **for** j **in** range(0,r):  
 z += f / x  
 f = f \* -1  
 x = x + 2  
  
 print(z)  
main()

This is the best I could do but I couldn’t figure out why this doesn’t work mathematically