Exercise 1

14/14 points (graded)

**ESTIMATED TIME TO COMPLETE: 10 minutes**

Below are some short Python programs. For each program, answer the associated questions.

Try to answer the questions without running the code. Check your answers, then run the code for the ones you get wrong.

iteration = 0

count = 0

while iteration < 5:

# the variable 'letter' in the loop stands for every

# character, including spaces and commas!

for letter in "hello, world":

count += 1

print("Iteration " + str(iteration) + "; count is: " + str(count))

iteration += 1

* 1. What is the value of the variable count that is printed out (at the print statement) on iteration 0?



* 1. What is the value of the variable count that is printed out (at the print statement) on iteration 1?



* 1. What is the value of the variable count that is printed out (at the print statement) on iteration 2?



* 1. What is the value of the variable count that is printed out (at the print statement) on iteration 3?



* 1. What is the value of the variable count that is printed out (at the print statement) on iteration 4?



1. iteration = 0
2. while iteration < 5:
3. count = 0
4. for letter in "hello, world":
5. count += 1
6. print("Iteration " + str(iteration) + "; count is: " + str(count))
7. iteration += 1
   1. What is the value of the variable count that is printed out (at the print statement) on iteration 0?



* 1. What is the value of the variable count that is printed out (at the print statement) on iteration 1?



* 1. What is the value of the variable count that is printed out (at the print statement) on iteration 2?



* 1. What is the value of the variable count that is printed out (at the print statement) on iteration 3?



* 1. What is the value of the variable count that is printed out (at the print statement) on iteration 4?



1. iteration = 0
2. while iteration < 5:
3. count = 0
4. for letter in "hello, world":
5. count += 1
6. if iteration % 2 == 0:
7. break
8. print("Iteration " + str(iteration) + "; count is: " + str(count))
9. iteration += 1
   1. How many times will the print statement be executed?



* 1. What is the largest value of the variable iteration that will be printed out (at the print statement)?



* 1. What is the largest value of the variable count that will be printed out (at the print statement)?



* 1. What is the smallest value of the variable count that will be printed out (at the print statement)?



Finger Exercises due Mar 31, 2021 16:30 PDT

Completed

### Exercise 2

3/3 points (graded)

**ESTIMATED TIME TO COMPLETE: 10 minutes**

Consider the following code:

x = 25

epsilon = 0.01

step = 0.1

guess = 0.0

while guess <= x:

if abs(guess\*\*2 -x) < epsilon:

break

else:

guess += step

if abs(guess\*\*2 - x) >= epsilon:

print('failed')

else:

print('succeeded: ' + str(guess))

If this code is executed, it will print succeeded: 4.9999999999998 (or succeeded: 5.0). Remember floating point errors?

Now suppose we try the following:

x = 25

epsilon = 0.01

step = 0.1

guess = 0.0

while guess <= x:

if abs(guess\*\*2 -x) >= epsilon:

guess += step

if abs(guess\*\*2 - x) >= epsilon:

print('failed')

else:

print('succeeded: ' + str(guess))

Select the answer that best describes what occurs when the above code is executed:

Script successfully completes, and prints out succeeded: 4.9999999999998 (or succeeded: 5.0)

Script successfully completes, but prints out failed

Script successfully completes, but prints out succeeded: followed by some number not really close to 5.0

Script enters an infinite loop and never terminates

correct

Now suppose we try

x = 25

epsilon = 0.01

step = 0.1

guess = 0.0

while abs(guess\*\*2-x) >= epsilon:

if guess <= x:

guess += step

else:

break

if abs(guess\*\*2 - x) >= epsilon:

print('failed')

else:

print('succeeded: ' + str(guess))

Select the answer that best describes what occurs when the above code is executed:

Script successfully completes, and prints out succeeded: 4.9999999999998 (or succeeded: 5.0)

Script successfully completes, but prints out failed

Script successfully completes, but prints out succeeded: followed by some number not really close to 5.0

Script enters an infinite loop and never terminates

correct

Finally, let's use the same code as immediately above, but change the first line to x = 23. Note that the square root of 23 is roughly 4.7958.

Select the answer that best describes what occurs when the modified code is executed:

Script successfully completes, and prints out succeeded: 4.9999999999998 (or succeeded: 5.0)

Script successfully completes, but prints out failed

Script successfully completes, but prints out succeeded: followed by some number not really close to 5.0

Script enters an infinite loop and never terminates

Exercise 3

4/4 points (graded)

**ESTIMATED TIME TO COMPLETE: 5 minutes**

1. True or False? The internal computer representation of any number is always an approximation.

True

False

1. The decimal 11 is what binary?:

11

1011

1101

cannot be converted

1. True or False? The internal representation of the decimal number 1/10 = 0.1 requires an infinite number of digits.

True

False

1. After many computations, you get two floating numbers stored in variables a and b. Your code compares the numbers with a == b.

Doing the comparison will always lead to a correct program.

Doing the comparison will sometimes lead to a correct program.

Doing the comparison will never lead to a correct program.