*Python Midterm Review Part 2*

1. How many times will the following code print "Welcome to Python"?

count = 0

while count < 10:

print("Welcome to Python")

count += 1

* 1. 8
  2. 9
  3. **10**
  4. 11
  5. 0

1. Analyze the following code. (More than one answer)

count = 0

while count < 100:

print ("Welcome to Python!") #Point A

count += 1 #Point B

#Point C

* 1. **count < 100 is always True at Point A**
  2. count < 100 is always True at Point B
  3. count < 100 is always False at Point B
  4. count < 100 is always True at Point C
  5. **count < 100 is always False at Point C**

1. How many times will the following code print "Welcome to Python" if count = 0?

while count < 10:

print ("Welcome to Python")

* 1. 8
  2. 9
  3. **10**
  4. 11
  5. infinite number of times

1. What will be displayed when the following code is executed if number = 6?

while number > 0:

number -= 3

print (number, end = ' ')

* 1. 6 3 0
  2. 6 3
  3. **3 0**
  4. 3 0 -3
  5. 0 -3

1. Analyze the following statement if sum = 0:

for d in range (0, 10, 0.1):

sum += sum + d

* 1. The program has a syntax error because the range function cannot have three arguments
  2. **The program has a syntax error because the arguments in the range must be integers**
  3. The program runs in an infinite loop
  4. The program runs fine

1. Which of the following loops prints "Welcome to Python" 10 times?

**A:**

for count in range(1, 10):

print ("Welcome to Python")

**B:**

for count in range(0, 10):

print ("Welcome to Python")

**C:**

for count in range(1, 11):

print ("Welcome to Python")

**D:**

for count in range(1, 12):

print ("Welcome to Python")

* 1. BD
  2. ABC
  3. AC
  4. **BC**
  5. AB

1. Which of the following loops correctly computes 1/2 + 2/3 + 3/4 + ... + 99/100?

**A:**

sum = 0

for i in range(1, 99):

sum += i / (i + 1)

print ("Sum is", sum)

**B:**

sum = 0

for i in range(1, 100):

sum += i / (i + 1)

print ("Sum is", sum)

**C:**

sum = 0

for i in range(1.0, 99.0):

sum += i / (i + 1)

print ("Sum is", sum)

**D:**

sum = 0

for i in range(1.0, 100.0):

sum += i / (i + 1)

print ("Sum is", sum)

1. BCD
2. ABCD
3. **B**
4. CDE
5. CD
6. Given the following four patterns,

Pattern A Pattern B Pattern C Pattern D

1 1 2 3 4 5 6 1 1 2 3 4 5 6

1 2 1 2 3 4 5 2 1 1 2 3 4 5

1 2 3 1 2 3 4 3 2 1 1 2 3 4

1 2 3 4 1 2 3 4 3 2 1 1 2 3

1 2 3 4 5 1 2 5 4 3 2 1 1 2

1 2 3 4 5 6 1 6 5 4 3 2 1 1

Which of the pattern is produced by the following code?

for i in range(1, 6 + 1):

for j in range(6, 0, -1):

print(j if j <= i else " ", end = " ")

print()

* 1. Pattern A
  2. Pattern B
  3. Pattern C
  4. Pattern D
  5. **??**

1. Will the following program terminate?

balance = 10

while True:

if balance < 9: break

balance = balance – 9

* 1. Yes
  2. **No**

1. What is sum after the following loop terminates?

sum = 0

item = 0

while item < 5:

item += 1

sum += item

if sum > 4:

item = 5

print(sum)

* 1. 5
  2. 6
  3. 7
  4. **8**

1. What will be displayed by after the following loop terminates?

number = 25

isPrime = True

i = 2

while i < number and isPrime:

if number % i == 0:

isPrime = False

i += 1

print("i is", i, "isPrime is", isPrime)

* 1. i is 5 isPrime is True
  2. **i is 5 isPrime is False**
  3. i is 6 isPrime is True
  4. i is 6 isPrime is False

1. What will be displayed by after the following loop terminates?

number = 25

isPrime = True

for i in range(2, number):

if number % i == 0:

isPrime = False

break

print("i is", i, "isPrime is", isPrime)

* 1. i is 5 isPrime is True
  2. **i is 5 isPrime is False**
  3. i is 6 isPrime is True
  4. i is 6 isPrime is False

1. What is the number of iterations in the following loop:

for i in range(1, n):

# something

* 1. 2\*n
  2. n
  3. **n – 1**
  4. n + 1

1. The header of a function consists of \_\_\_\_\_\_\_\_\_\_\_\_.
   1. function name
   2. **function name and parameter list**
   3. parameter list
2. A function \_\_\_\_\_\_\_\_\_\_.
   1. Must have at least one parameter
   2. **May have no parameters**
   3. Must always have a return statement to return a value
   4. Must always have a return statement to return multiple value
3. Arguments to functions always appear within \_\_\_\_\_\_\_\_.
   1. brackets
   2. **parentheses**
   3. curly braces
   4. quotation marks
4. Does the function call in the following function cause syntax errors?

import math

def circle\_area():

print(math.pi \* radius\*\*2)

def main():

radius = int(input("what is the radius? "))

circle\_area()

main()

* 1. **Yes**
  2. No

1. Consider the following incomplete code:

def f(number):

# Missing function body

print(f(5))

The missing function body should be \_\_\_\_\_\_\_\_.

* 1. return "number"
  2. print(number)
  3. print("number")
  4. **return number**

1. Given the following function

def nPrint(message, n):

while n > 0:

print (message, end=’’)

n -= 1

What will be displayed by the call nPrint ('a', 4)?

* 1. aaaaa
  2. **aaaa**
  3. aaa
  4. invalid call
  5. infinite loop

1. What will be displayed by the following code if x = 1?

def f1():

x = x + 2

print(y, end=’ ‘)

f1()

print(x)

* 1. 1 3
  2. **3 1**
  3. The program has a runtime error because x is not defined
  4. 1 1
  5. 3 3

1. What will be displayed by the following code if x = 1?

def f1():

x = 3

print(x, end= ‘ ‘)

f1()

print(x)

* 1. 1 3
  2. **3 1**
  3. The program has a runtime error because x is not defined
  4. 1 1
  5. 3 3

1. What will be displayed by the following code if x = 1?

def f1():

x = x + 2

print(x, end= ‘ ‘)

f1()

print(x)

* 1. 1 3
  2. **3 1**
  3. The program has a runtime error because x is not defined
  4. 1 1
  5. 3 3

1. What will be displayed by the following code if x = 1?

def f1():

global x

x = x + 2

print(x, end= ‘ ‘)

f1()

print(x)

* 1. 1 3
  2. 3 1
  3. The program has a runtime error because x is not defined
  4. 1 1
  5. **3 3**

1. What is len("Good")?
   1. 1
   2. 2
   3. 3
   4. **4**
   5. -1
2. What is "Programming is fun"[4: 6]?
   1. ram
   2. **ra**
   3. r
   4. pr
   5. pro

1. What is "Programming is fun"[-1]?
   1. Pr
   2. P
   3. Fun
   4. **N**
   5. un

1. What is "Programming is fun"[1:1]?
   1. P
   2. R
   3. Pr
   4. ''
   5. **incorrect expression**

1. \_\_\_\_\_\_\_\_\_\_ creates a list.
   1. **list1 = list()**
   2. list1 = {}
   3. list1 = list([12, 4, 4])
   4. **list1 = [12, 4, 4]**
   5. **list1 = [1, "3", "red"]**

1. What is list ("abcd")?
   1. ['a', 'b', 'c', 'd']
   2. ['ab']
   3. ['cd']
   4. **['abcd']**
2. Suppose list1 is [3, 4, 5, 20, 5, 25, 1, 3], what is len(list1)?
   1. 6
   2. 7
   3. **8**
   4. 5
   5. 4