### 1. What is the output of the following snippet?

l1 = [1,2]

for v in range(2):

l1.insert(-1,l1[v])

print(l1)

1. **[1, 2, 2, 2]**
2. **[1, 1, 1, 2]**
3. **[1, 2, 1, 2]**
4. **[1, 1, 2, 2]**

### 2. The meaning of a positional argument is determined by:

1. **the argument’s name specified along with its value**
2. **its connection with existing variables**
3. **its value**
4. **its position within the argument list**

### 3. Which of the following sentences are true? Choose all that apply.

nums = [1,2,3]

vals = nums

1. **vals is longer than nums**
2. **nums is longer than vals**
3. **nums and vals are different names of the ame list**
4. **nums and vals are different lists**

### 4. An operator able to check whether two values are not equal is coded as:

1. **not ==**
2. **<>**
3. **!=**
4. **=/=**

### 5. The following snippet:

def func1(a):

return None

def func2(a):

return func1(a)\*func1(a)

print(func2(2))

1. **will output 2**
2. **will cause a runtime error**
3. **will output 4**
4. **will output 16**

### 6. The result of the following division:

1 // 2

1. **cannot be predicted**
2. **is equal to 0.5**
3. **is equal to 0.0**
4. **is equal to 0**

### 7. The following snippet:

def func(a,b):

return b \*\* a

print(func(b=2,2))

1. **will output 4**
2. **is erroneous**
3. **will return None**
4. **will output 2**

### 8. What value will be assigned to the x variable?

z = 0

y = 10

x = y < z and z > y or y > z and z < y

* 1. **0**
  2. **False**
  3. **1**
  4. **True**

### 9. One of the following variables’ names is illegal – which one?

1. **in\_**
2. **IN**
3. **in**
4. **In**

### 10. What is the output of the following snippet?

list = [x\*x for x in range(5)]

def fun(L):

del L[L[2]]

return L

print(fun(list))

1. **[0, 1, 4, 16]**
2. **[1, 4, 9, 16]**
3. **[0, 1, 4, 16]**
4. **[0, 1, 4, 9]**

### 11. What is the output of the following piece of code?

x=1

y=2

x, y, z = x, x, y

z, y, z = x, y, z

print(x,y,z)

1. **1 1 2**
2. **1 2 2**
3. **2 1 2**
4. **1 2 1**

### 12. What will the output of the following snippet?

a = 1

b = 0

a = a ^ b

b = a ^ b

a = a ^ b

print(a,b)

1. 0 1
2. 1 0
3. 0 0
4. 1 1

### 13. What is the output of the following snippet?

def fun(x):

if x % 2 == 0:

return 1

else:

return 2

print(fun(fun(2)))

1. **None**
2. **1**
3. **the code will cause a run-time error**
4. **2**

### 14. Take a look at the snippet and choose the true statement:

nums = [1,2,3]

vals = nums

del vals[:]

1. **nums and vals are different names of the same list**
2. **vals is longer than nums**
3. **the snippet will cause a runtime error**
4. **nums and vals are different lists**

### 15. What is the output of the following piece of code if the user enters two lines containing 3 and 2 respectively?

x=int(input())

y=int(input())

x = x % y

x = x % y

y = y % x

print(y)

1. **2**
2. **1**
3. **0**
4. **3**

### 16. What is the output of the following piece of code if the user enters two lines containing 3 and 6 respectively?

y=input()

x=input()

print(x+y)

1. **6**
2. **3**
3. **36**
4. **63**

### 17. What is the output of the following piece of code?

print(“a”,”b”,”c”,sep=”sep”)

1. **abc**
2. **asepbsepcsep**
3. **asepbsepc**
4. **a b c**

### 18. What is the output of the following piece of code?

X = 1 // 5 + 1 / 5

print(X)

1. **0.4**
2. **0.0**
3. **0.2**
4. **0**

### 19. Assuming that the tuple is a correctly created tuple, the fact that tuples are immutable means that the following instruction:

tuple[1] = tuple[1] + tuple[0]

1. **is fully correct**
2. **is illegal**
3. **may be illegal if the tuple contains strings**
4. **can be executed if and only if the tuple contains at least two elements**

### 20. What is the output of the following piece of code if the user enters two lines containing 2 and 4 respectively?

x=float(input())

y=float(input())

print(y \*\* (1/x))

1. **4.0**
2. **2.0**
3. **1.0**
4. **0.0**

21. What is the output of the following snippet?

dct = { ‘one’:’two’, ‘three’:’one’, ‘two’:’three’ }

v = dct[‘three’]

for k in range(len(dct)):

v = dct[v]

print(v)

1. **two**
2. **three**
3. **one**
4. **(‘one’, ‘two’, ‘three’)**

### 22. How many elements does the L list contain?

L = [i for i in range(-1,-2)]

1. **1**
2. **2**
3. **0**
4. **3**

### 23. Which of the following lines improperly invokes the function defined as:

def fun(a,b,c=0)

Choose all that apply.

1. **fun(b=1):**
2. **fun(a=1,b=0,c=0):**
3. **fun(a=0,b=0):**
4. **fun(0,1,2):**

### 24. What is the output of the following snippet?

def fun(x,y):

if x == y:

return x

else:

return fun(x,y-1)

print(fun(0,3))

1. **0**
2. **1**
3. **the snippet will cause a runtime error**
4. **2**

### 25. How many stars will the following snippet send to the console?

i = 0

while i < i + 2 :

i += 1

print(“\*”)

else:

print(“\*”)

1. **the snippet will enter an infinite loop**
2. **zero**
3. **one**
4. **two**

### 26. What is the output of the following snippet?

tup = (1, 2, 4, 8)

tup = tup[-2:-1]

tup = tup[-1]

print(tup)

1. **4**
2. **(4)**
3. **44**
4. **(4,)**

### 27. What is the output of the following snippet?

dd = { “1”:”0″, “0”:”1″ }

for x in dd.vals():

print(x,end=””)

1. **1 0**
2. **the code is erroneous**
3. **0 0**
4. **0 1**

### 28. What is the output of the following snippet?

dct = {}

dct[‘1’] = (1,2)

dct[‘2’] = (2,1)

for x in dct.keys():

print(dct[x][1],end=””)

1. **21**
2. **(1,2)**
3. **(2,1)**
4. **12**

### 29. What is the output of the following snippet?

def fun(inp=2,out=3):

return inp \* out

print(fun(out=2))

1. **2**
2. **the snippet is erroneous**
3. **6**
4. **4**

### 30. How many hashes will the following snippet send to the console?

lst = [[x for x in range(3)]

for y in range(3)]

for r in range(3):

for c in range(3):

if lst[r][c] % 2 != 0:

print(“#”)

1. **zero**
2. **three**
3. **nine**
4. **six**