(1-6) Given the following Python code:

```python
>>> lake = 'Odell'
```

What is the result when the following Python code is executed?

1) ```>>> lake```  
   a) 'Odell'  
   b) 'lake'  
   c) lake  
   d) NameError  
   e) Odell  

2) ```>>> print(lake)```  
   a) 'Odell'  
   b) 'lake'  
   c) lake  
   d) NameError  
   e) Odell  

3) ```>>> 'lake'```  
   a) 'Odell'  
   b) 'lake'  
   c) lake  
   d) NameError  
   e) Odell  

4) ```>>> len('lake')```  
   a) 3  
   b) 4  
   c) 5  
   d) 6  
   e) NameError  

5) ```>>> len(lake)```  
   a) 3  
   b) 4  
   c) 5  
   d) 6  
   e) NameError  

6) ```>>> lake[1]```  
   a) 'Odell'  
   b) NameError  
   c) 'O'  
   d) IndexError  
   e) 'd'  

7) What will be printed when the following Python code is executed in the Shell?

   ```>>> pow```  
   a) pow  
   b) <built-in function pow>  
   c) NameError  
   d) <class 'int'>  
   e) 'pow'  

8) What is the result when the following Python code is executed?

   ```>>> len(97403)```  
   a) TypeError  
   b) NameError  
   c) IndexError  
   d) 4  
   e) 5
(9-10) Given the following Python code:

```python
>>> greeting = 'hi'
```

What is the result when the following Python code is executed?

9) ```
```

a) greeting  
b) 'hi'  
c) True  
d) False  
e) SyntaxError

10) ```
```

a) greeting  
b) 'hi'  
c) True  
d) False  
e) SyntaxError

11) When the following Python code is executed, what will be printed?

```python
nextclass = '123'
for ch in nextclass:
    print(ch*2, end='')
```

a) 0123  
b) 123  
c) 12  
d) 1233  
e) 112233

12) When the following Python code is executed, what will be printed?

```python
>>> days_in_month = input()
30
>>> type(days_in_month)
```

a) int  
b) float  
c) str  
d) list  
e) function

13) What will be the value of `ttlc` after the following Python code is executed?

```python
r = 4
g = r // 2
p = r + g
ttlc = r + g + p
```

a) 12  
b) 12.0  
c) 0  
d) 1  
e) 1.0
Given the following Python code:

```python
from turtle import *
def q14(n):
    '''(int) -> None
    Exam function.'''
    le = 100
    angle = 360 / n
    for i in range(n):
        fd(le)
        lt(angle)
    return None

>>> q14(4)
```

14) At line 8 the value of n is
   a) undefined       b) 4       c) 4.0       d) 100

15) After lines 8 and 9 execute, the values of le and angle are
   a) undefined       b) 100, 90       c) 100.0, 90.0       d) 100, 90.0

16) After line 10 executes the first time, the value of i is
   a) undefined       b) 0       c) 1       d) 0.0       e) 1.0

17) When the function is done executing,
   a) a drawing of a square is returned       b) None and a drawing of a square are returned
   c) None is returned and a square is drawn as a side effect       d) function is an infinite loop

18) To implement forward movement for a variety of lengths, an additional parameter (le) is added to the function header. What other lines in the function would need to be changed?
   a) No other lines       b) 4, 8, 11, 12       c) 4, 8, 11       d) 4, 8       e) 4, 11

19) What will be the result when the following Python code is executed?

```python
bill = 100.00
tip18 = bill * .18
tip20 = bill * .20
print(tip18, tip20)
```

   a) 18 20       b) 18.0 20       c) 18 20.0       d) 18.0 20.0
(20-22) Given the following Python code:

```python
1 def q20(w, x):
2     ''' (int, int) -> int
3     Exam function.
4     '''
5     w = 200
6     x = 300
7     result = w + x
8     return result
```

20) What is the result of executing the following Python code?

```python
>>> q20(2, 3)
```

a) undefined  b) 5  c) 500  d) NameError  e) TypeError

21) What is the result of executing the following Python code?

```python
>>> q20(3, 4)
```

a) 5  b) 7  c) 500  d) NameError  e) TypeError

22) What changes to function q20 are needed for it to return different results for different argument inputs?

a) change lines 1, 2  b) delete lines 6, 7  c) change lines 8, 9  d) delete lines 8, 9
Given the following Python code:

```python
import random

def q23():
    '''() -> (?)
    
    Exam function
    '''
    n = random.randint(1, 290)
    return n

def q24():
    '''() -> (?)
    
    Exam function
    '''
    n = random.randint(-178, 1)
    return n

def q25():
    '''() -> None
    
    Exam function
    '''
    wc_data = q23()
    temp_data = q24()

    print(wc_data, '\t', temp_data, '\t')
    return None

>>> q25()

23) What value is returned?

a) None b) -178 c) 1 d) 290 e) 1

24) What are possible values for wc_data and temp_data at line 24?

a) undefined b) 5, 0 c) 5.0, 0.0 d) 100, 100 e) True, True

25) Complete the type contracts (right hand side) for functions q23 and q24:

a) int, int b) int, float c) float, int d) None, None e) Boolean, Boolean
(26-28) Given the following Python code:

```python
1 def q26(x):
2     '''() -> (??)
3     Exam function
4     '''
5     result = x * x
6     return result
7
8 def q27(x):
9     '''() -> (??)
10    Exam function
11    '''
12     x += 1
13     return x
14
15 def main():
16     '''() -> None
17     Exam function
18     '''
19     result = q27(q26(3))
20     print(result)
21     return None
22
23 >>> main()
```

26) What is the value of `x` after line 6 is executed?

a) 3      b) 6      c) 9      d) undefined

27) What is the value of `x` after line 14 is executed?

a) 3      b) 4      c) 9      d) 10      e) undefined

28) What is printed in the Shell?

a)       b)       c)       d)       e)       
9        9        10       None       nothing is printed
10
None
After executing the following Python code:

```python
1 def q29(weight):
2     '''(number) -> (??)
3     Exam function.
4     '''
5     cost = weight * .05
6     if weight > 100:
7         cost = cost - 1.50
8     return cost

>>> q29(100)
```

29) How many times was line 8 executed?

a) 0   b) 1   c) 2   d) 0 or 1 or 2   e) cannot be determined

30) Complete the type contract:

a) number   b) int    c) float   d) str   e) None

31) Choose the best set of test cases for q29:

a) q29(0)   b) q29(10)   c) q29(0)
   q29(1)   q29(20)   q29(1)
   q29(20)   q29(30)   q29(2)
   q29(100)   q29(40)   q29(3)
   q29(200)   q29(50)   q29(4)

32) Which line(s) of code need(s) to be changed for the discount to apply to weights of 100 or more?

a) 7   b) 2,7   c) 7,8   d) 6,7,8,10   e) no code changes are needed

33) What is the result when the following Python code is executed?

```python
>>> s = 'University of Oregon'
>>> s[s.find('O')]:]
```

a) 14   b) 'UO'   c) 'Oregon'   d) ''   e) 'O'
(34) Given the following Python code:

```python
if a > b:
    c = 10
elif a < b:
    c = 15
else:
    c = 20
print(c)
```

If after the code is executed, the value 20 has been printed, what must be True?

a) a > b  

b) a < b  

c) a == b  

d) c > b  

e) c > a

(35-37) After executing the following Python code:

```python
def q35(s1, s2):
    '''(str, str) -> ??
    Exam function
    ...
    field_width = 18
    ttl_len = len(s1) + len(s2)
    fits = ttl_len <= field_width
    return fits

>>> q35('Hood', 'Sandy')
```

35) Complete the type contract:

a) str  
b) int  
c) float  
d) Boolean  
e) None

36) What is printed in the Shell?

a) 18  
b) 9  
c) True  
d) False  
e) nothing is printed

37) Which line(s) of code need(s) to be changed such that the field width is specified at function call time?

a) 1,2  
b) 2,8  
c) 1,2,8  
d) 1,2,8,9  
e) 1,2,6
(38) Given the following Python code:

```python
def any_upper(astring):
    '''(str) -> Boolean

    Returns True if there is any upper-case character in astring, a string of alphabetic characters. Otherwise return False.
    '''
    for ch in astring:
        if ch.isupper():
            return True
        else:
            return False
```

Executing any_upper will <??> return the correct result.

a) always  b) sometimes  c) never  d) cannot be determined

(39-40) Given the following UNTESTED Python code:

```python
def calc_avg(dataset):
    '''(string) -> float

    returns average of values in input string values, but zeros do not count at all
    
    >>> calc_avg('23')
    2.5
    >>> calc_avg('203')
    2.5
    '''
    count = 0
    total = 0
    for n in dataset:
        if n != '0':
            total += int(n)
            count += 1
    avg = total / count
    return avg
```

When the following code is executed

```python
>>> calc_avg('303')
```
39) \( n \) will have the value(s)

a) '3', '4', '5'  
   b) '3'  
   c) '3', '0', '3'  
   d) '3', '3'  
   e) '1', '2', '3'

40) What value will be returned?

```python
>>> calc_avg('303')
```

a) 2.0  
   b) 2.5  
   c) 3.0  
   d) None

(41) What is the result when the following Python code is executed?

```python
>>> uli = ['University of Oregon', 'Cal', 'Oregon State']
>>> uli.sort()
>>> uli
```

a) ['University of Oregon', 'Cal', 'Oregon State']  
   b) ['Oregon State', 'Cal', 'University of Oregon']  
   c) ['Cal', 'Oregon State', 'University of Oregon']  
   d) []  
   e) None

(42) What is the result when the following Python code is executed?

```python
>>> uli = ['University of Oregon', 'Cal', 'Oregon State']
>>> uli = uli.sort()
>>> uli
```

a) ['University of Oregon', 'Cal', 'Oregon State']  
   b) ['Oregon State', 'Cal', 'University of Oregon']  
   c) ['Cal', 'Oregon State', 'University of Oregon']  
   d) []  
   e) None
(43-44) Given the following Python code to process a text file:

def processf(fname):
    '''(??) -> ??

    Exam function
    '''
    with open(fname, 'r') as examf:
        ??

    return None

43) Complete the type contract:

a) str, str  b) None, None  c) str, None  d) int, None  e) list, None

44) If `fname` is a text file where the first line is a header that identifies what kind of file it is, what code should be included in the function to move the file pointer past the file header?

a) examf.read()

b) examf.readline()

c) examf.readlines()

d) examf.write()

45) What will be the result of executing the following Python code:

def best(greeting):
    '''(str) -> None

    print a greeting
    '''
    start = greeting.find('Best')
    print(greeting[start:]

    return None

>>> best('CIS122: Best wishes for a pleasant summer.')</n
a) Best wishes for a pleasant summer.

b) Best wishes for a pleasant summer.

c) Best wishes for a pleasant summer.

d) Best wishes for a pleasant summer.

e) Best wishes for a pleasant summer.