What is the result when the following Python code is executed (in order, in one Shell session):

1. >>> 9 / 2
   a) 4.5           b) 4.0           c) 4            d) 1            e) TypeError

2. >>> 9 // 2
   a) 4.5           b) 4.0           c) 4            d) 1            e) TypeError

3. >>> 9 % 2
   a) 4.5           b) 4.0           c) 4            d) 1            e) TypeError

4. >>> 'hello'
   a) 'hello'       b) <class 'str'>  c) TypeError    d) NameError

5. >>> greeting
   a) 'hello'       b) <class 'str'>  c) TypeError    d) NameError

6. >>> greeting = 'hello'
       >>> greeting
   a) 'hello'       b) <class 'str'>  c) TypeError    d) NameError

7. >>> type(False)
   a) <class 'int'>  b) <class 'float'>  c) <class 'bool'>
      d) <class 'str'>  e) <class 'python'>

8. >>> type(abs(7))
   a) <class 'int'>  b) <class 'float'>  c) <class 'bool'>
      d) <class 'str'>  e) <class 'function'>

9. >>> isinstance(101, float) == True
   a) True          b) 'True'         c) true       d) False       e) 'false'

10. >>> isinstance(101, float)
    a) True         b) 'True'        c) true       d) False       c) 'false'
11-13. Given the following Python code:

```python
>>> x = 'CIS 210'
>>> id(x)
4391509160
>>> y = x
>>> id(y)
??-1
>>> x = 'the end'
>>> id(x)
??-2
>>> y
??-3
```

11. 4391509160 refers to a(n)

   a) assignment statement       b) function       c) `None` type       d) keyword       e) memory location

12. The value printed at ??-1 will also be 4391509160 (yes or no); the value printed at ??-2 will also be 4391509160 (yes or no).

   a) yes/yes     b) no/no     c) yes/no     d) no/yes

13. The value printed at ??-3 will be

   a) 4391509160     b) 'CIS 210'     c) 'the end'     d) `None`

14. The following Python code

```python
>>> x = 'hi'
>>> x = 0
>>> x = x < 0
```

is an example of what feature of Python?

   a) strong typing       b) **dynamic typing**       c) operator overloading       d) loops       e) conditionals

15. The following Python code

```python
>>> x = 'hi' + '-' + 'bye'
>>> y = 99 + 100
```

demonstrates what feature of Python?

   a) strong typing       b) dynamic typing       c) **operator overloading**       d) weak typing       e) static typing
16. The following Python code

```python
>>> x = 'hi' + 99
Traceback (most recent call last):
  File "<pyshell#41>" , line 1 , in <module>
    x = 'hi' + 99
TypeError: must be str, not int
```

is an example of what feature of Python?

a) strong typing  
b) dynamic typing  
c) operator overloading  
d) weak typing  
e) static typing  

17. Given the following:

1 - >>> ftemp = 212
2 - >>> ctemp = (ftemp - 32) * 5/9
3 - >>> ctemp = ftemp - 32 * 5/9

The value of `ctemp` will [??] from line 2 to line 3; the id of `ctemp` will [??] from line 2 to line 3

a) stay the same/change  
b) change/stay the same  
c) stay the same/stay the same  
d) change/change

18-19. What will be printed when the following Python code is executed?

```python
n = 5
mysum = 0
for ctr in range(1, n):
    myctr = mysum + ctr
print(mysum)
```

18.  
a) 0  
b) 1  
c) 10  
d) 15  
e) nothing will be printed  

19. This code does not work as intended. This bug may be attributed to Python's

a) strong typing  
b) dynamic typing  
c) operator overloading  
d) weak typing  
e) static typing

20-21. What will be printed when the following Python code is executed?

```python
n = 5
mysum = 0
for ctr in range(1, n):
    mysum = mysum + ctr
print(mysum)
```
20. a) 0   b) 1   c) 10   d) 15   e) nothing will be printed

21. This code is an example of
   a) accumulator pattern   b) TypeError   c) conditional
   d) indefinite iteration   e) infinite loop

22. Given the following Python code:

0 import math

1 def isInCircle(x, y, r):
   2     '''(number, number, number) -> ??
   3         Returns True if point (x, y) is in
   4         the circle with radius r.
   5     >>> isInCircle(0, 0, 1)
   6         True
   7     >>> isInCircle(.5, .5, 1)
   8         True
   9     >>> isInCircle(1, 2, 1)
  10         False
  11         '''
  12     d = math.sqrt(x**2 + y**2)
  13     isIn = d <= r
  14     return isIn

22. Complete the type contract:

   a) number   b) int   c) float   d) bool   e) str

23. Which code would give the same results as isInCircle lines 12-14 (changes are in bold)?

   a) d = math.sqrt(x**2 + y**2)
      return d = r

   b) d = math.sqrt(pow(x, 2) + pow(y, 2))
      return d <= r

   c) d = math.sqrt(x**2 + y**2)
      return d < r

   d) d = math.sqrt(pow(x, 2) + pow(y, 2))
      isIn = d < r
      return isIn

   e) d = math.sqrt(x**2 + y**2)
      return d

24-28. Given the following Python code:
def q24(s):
    
    Test function.

    >>> q24('The quick brown fox')
    
    >>> q24('Hello, world."
    
    result = 999
    for i in range(len(s)):
        if s[i] == 'E' or s[i] == 'e':
            result = i
    return result

q24('Hello')

24. Complete the type contract for q24:

   a) str/str     b) int/int     c) str/float    d) str/bool     e) str/int

25. Executing this function will

   a) Return the number of occurrences of 'e' in s, or 999 if none.

   b) Return the number of occurrences of 'E' in s, or 999 if none.

   c) Return the sum of a) and b), or 999 if none.

   d) Return the position of the first occurrence of 'e' or 'E' in s, or
      999 if none.

   e) Return the position of the last occurrence of 'e' or 'E' in s, or
      999 if none.

26. The first time the for loop executes, the value of i is

   a) 'H'        b) 0          c) 1          d) 4          e) 5

27. The first time the for loop executes, the value of s[i] == 'E' or s[i] == 'e' is

   a) 'E'        b) 'e'        c) True       d) False      e) 'False'

28. To determine this value, Python evaluated

   a) b) c) d) 
   s[i] == 'E'  s[i] == 'E'  s[i] == 'e'   result += 1  
   s[i] == 'e'
29. Given the following Python code:

```python
def q29(s1):
    '''(str) -> str

    s2 = ''
    for ch in s1:
        if ch not in s2:
            s2 += ch

    return s2
```

Which brief description is appropriate for `q29`?

a) copies `s1` to `s2`; returns `s2`
b) copies all characters except the last character in `s1` to `s2`; returns `s2`
c) copies 1st occurrence of each character in `s1` to `s2`; returns `s2`
d) determines whether `s1` is an empty string
e) creates and returns `s2`, a string of the characters that repeat (occur more than once) in `s1`

30. Given the following Python code:

```python
def q30(score):
    ''' exam function '''

    gradepoint = 0
    if score >= 90:
        gradepoint = 4
    elif score >= 80:
        gradepoint = 3
    elif score >= 70:
        gradepoint = 2
    elif score >= 60:
        gradepoint = 1

    return gradepoint
```

What is the result of executing `>>> q30(80)`?

a) 4     b) 3     c) 2     d) 1     e) NameError
31. Write a Python function, `fb`, that implements fizzbuzz.

Recall that fizzbuzz is played as follows: starting at 1, count up to some number, `n` (inclusive). For each number, if it is divisible by 3, say (display) “fizz”. If it is divisible by 5, display “buzz”. If it is divisible by both 3 and 5, display “fizzbuzz”. Otherwise, just display the number. Function `fb` should have one parameter, `n`, which is the number that will end the game.

For each round of fizzbuzz, the function should print the number or “fizz” or “buzz” or “fizzbuzz”. At the end of the game, print “Game over!” `fb` should return the `None` value.

For example,

```python
>>> fb(4)
1
2
fizz
4
Game over!
```

Function `fb` should include a complete docstring: type contract, (very) brief description, and one simple example of a function call.

```python
def fb(n):
    '''(int) -> None
    Play fizzbuzz up to n.
    Results are printed during play;
    None value is returned
    
    >>> fizzbuzz(15)
    1
    2
    fizz
    4
    buzz
    fizz
    ...
    fizzbuzz
    '''
    for i in range(1, n+1):
        m3 = (i % 3) == 0
        m5 = (i % 5) == 0
        if m3 and m5:
            print('fizzbuzz')
        elif m3:
            print('fizz')
        elif m5:
            print('buzz')
        else:
            print(i)
    print('Game over!')
    return None
```