FALL 2017 - COMP 141
MIDTERM 1 PRACTICE PROBLEMS

1. The _____ function reads a piece of data that has been entered at the keyboard and returns that piece of data, as a string, back to the program.
   a. input
   b. output
   c. eval_input
   d. string_input

2. In a print statement, you can set the _____ argument to a space or empty string to stop the output from advancing to a new line.
   a. stop
   b. end
   c. separator
   d. newline

3. After the execution of the following statement, the variable **sold** will reference the numeric literal value as a(n) _____ data type: sold = 256.752
   a. int
   b. float
   c. str
   d. currency

4. After the execution of the following statement, the variable **price** will reference the value _____. price = int(68.549)
   a. 68
   b. 69
   c. 68.55
   d. 68.54

5. The _____ design technique can be used to break down an algorithm into functions.
   a. subtask
   b. block
   c. top-down
   d. simplification

6. The _____ of a local variable is the function in which the variable is created.
   a. global
   b. defined
   c. local
   d. scope

Questions 7-10 refer to the following code (line numbers present for reference).

```
1  def average(first, second, third):
2      avg = (first + second + third) / 3
3      print("average is", avg)
4
5  def main():
6      x = float(input("First number? "))
7      y = float(input("Second number? "))
8      z = float(input("Third number? "))
9      average(x, y, z)
10
11 main()
```

7. Line 1 is the function _____ for the function average.
   a. call
   b. header
   c. block
   d. parameter

8. In Lines 2 and 3, avg is a _____ variable to the function average.
   a. global
   b. constant
   c. defined
   d. local

9. In Line 1, first, second and third are _____ for the function average.
   a. headers
   b. returns
   c. parameters
   d. arguments

10. In Line 9, x, y, and z are _____ used when calling the average function from main.
   a. headers
   b. returns
   c. parameters
   d. arguments

11. It is recommended that programmers should avoid using _____ variables in a program when possible.
   a. local
   b. global
   c. string global
   d. keyword

12. What is the result of the following Boolean expression, if x equals 5, y equals 3, and z equals 8?

```
x < y or z > x
```

a. True
b. False
c. 8
d. 5

13. What is the result of the following Boolean expression, if x equals 5, y equals 3, and z equals 8?

```
not (x < y or z > x) and y < z
```

a. True
b. False
c. 8
d. 5

14. The expression `print(str(8) + str(9))` will output _____.

15. The result of the expression `12.3 + 6.7` is _____.

16. When applying the `.3f` formatting specifier to the following number, `76.15854`, the result is _____.

17. A(n) _____ statement will execute one block of statements if its condition is true, or another block if its condition is false.

18. A(n) _____-controlled loop causes a statement or set of statements to repeat as long as a condition is true.

19. _____ are notes of explanation that document lines or sections of a program.

20. What is x after the following statements?

```
x = 2
x *= x + 2
```

21. What is the output for y?

```
y = 0
for i in range(0, 10):
    y += i

print(y)
```

22. What will be displayed 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)
```

23. The following code displays _____.

```
age = 19

if age < 18:
    print("Minor")
elif age >= 18 and age < 65:
    print("Adult")
else:
    print("Senior Citizen")
```

24. Write code that will randomly generate a number between 0 and 100. If that number is greater than 50, output that it is “Too high”, otherwise, output “Too low”.

25. Write a function called `calculateAverage` that takes in three parameters `num1`, `num2` and `num3` and **`returns`** (not prints) the average of the three numbers.

26. Write a function called `compareNumbers` that takes in 2 parameters `num1` and `num2` and outputs the numbers in ascending order.

27. Given that n refers to a positive integer, write a loop to compute the sum of the squares of the first n counting numbers, and associate this value with total. Thus if n equals 4, your code should put 1*1 + 2*2 + 3*3 + 4*4 into total.

28. Write a loop that asks the user to enter a series of positive numbers. The user should enter a negative number to signal the end of the series. The program should output whether each number entered is even or odd.