

**Name:** \_\_\_\_\_

Use this quiz to help make sure you understand the videos/reading. **Answer all questions.** Make additional notes as desired. **Not sure of an answer?** Ask your instructor to explain in class and revise as needed then.

Throughout, where you are asked to “circle your choice”, you can circle or underline it (whichever you prefer).

**Textbook Reading:** Sections 3.1 – *The if Statement* (pages 92 – 97) and 3.2 – *Relational Operators* (pages 97 – 102)

1. Consider the following line of code, and add code after it that prints “Banana!” if the user inputs a number bigger than 20: [Hint: Just use an *if* statement here – no *elif* or *else*.]

```
n = int(input('Enter an integer: '))
```

2. Consider the following line of code, and add code after it that:

- prints “first!” if the first number that the user enters is bigger than the second number that the user enters,
- prints “second!” if the reverse is true, and
- prints “tie!” if the two numbers are the same.

```
a = int(input('Enter an integer: '))  
b = int(input('Enter a second integer: '))
```

3. Same as the previous question, but don't print anything at all in the case that the two numbers are the same. [Hint: You don't have to have an **else** clause if the logic of the problem does not demand it.]

4. Consider Versions 1 and 2 below.

**Version 1:**

```
if x > y:  
    print('good')  
else:  
    print('bad')
```

**Version 2:**

```
if x > y:  
    print('good')  
if x <= y:  
    print('bad')
```

**True or False:** Versions 1 and 2 are equivalent in their effect – that is, if Version 1 is given an x and y and Version 2 is given the same x and y, then they both print the same thing.

**True False** (circle your choice)

5. Which of the above Versions runs faster? (Circle your choice)

**Version 1**

**Version 2**

**Neither (they run equally fast)**

6. Which of the above Versions is clearer? (Circle your choice)

**Version 1**

**Version 2**

**Neither (they are equally clear)**

7. Which of the above Versions is better? (Circle your choice)

**Version 1**

**Version 2**

**Neither (they are equally good)**

**Video: The Accumulator Pattern – Part 2** [9:36 minutes]

8. What is the value of **x** after the following code executes: \_\_\_\_

```
x = 2
x = x + 1
x = x + 2
x = x + 3
```

9. What is the value of **y** after the following code executes: \_\_\_\_

```
y = 2
y = y * 2
y = y * 3
```

10. What are the values of **x**, **y** and **z** after the following code executes:

```
x = 1
y = 2
z = x + y
```

```
x = x + 2
y = y * 2
z = z + x + y
```

```
x = x + 3
y = y * 3
z = z + x + y
```

**x is** \_\_\_\_\_ **y is** \_\_\_\_\_ **z is** \_\_\_\_\_

11. Implement the following function (here, on paper – it is fine to make small errors):

```
def add_them(n):
    """ Returns 2 + 4 + 6 + 8 + 10 + ... + 2n """
```

12. Implement the following function (here, on paper – it is fine to make small errors):

```
def multiply_them(n):  
    """ Returns 2 * 4 * 6 * 8 * 10 * ... * 2n """
```

13. If you are doing a **counting** problem, what is a good choice for the name of the variable that does the counting: \_\_\_\_\_  
(Hint: **x** or non-descriptive names like it are poor choices.)

14. If you are doing a **summing** problem, what is a good choice for the name of the variable that does the summing: \_\_\_\_\_  
(Hint: **x** or non-descriptive names like it are poor choices.  
Also, **sum** is a poor choice because it happens to be the name of a built-in function.)

15. Suppose you wanted to count the number of integers from 3 to 40,000 that are prime. What **for ... in range ...** statement would you use? You do NOT have to provide the **body** of the **for** loop – JUST the **for ... in range ...** part. (Hint: There are at least two natural approaches to solving this problem; one of those runs twice as fast as the other.)

16. Implement the following function (here, on paper – it is fine to make small errors):

```
def count_them(m, n):  
    """  
    Returns the number of integers from m to n, inclusive,  
    whose cosine is less than 0.5.  
    Preconditions: m and n are integers with n >= m.  
    """
```

**Textbook Reading: Sections 3.3 – Nested Branches** (pages 106 - 109) and **3.4 – Multiple Alternatives** (pages 109 - 112)

17. In a game program, the scores of players A and B are stored in variables **scoreA** and **scoreB**. Assuming that the player with the larger score wins, write an **if/elif** sequence that prints out **'A won'** or **'B won'** or **'Game tied'**.

18. Beginners sometimes write statements such as the following:

```
if price > 100:
    discounted_price = price - 20
elif price <= 100:
    discounted_price = price - 10
```

Explain what change would improve this code.

19. **[Optional] Think about it:** You do not have to write anything for this problem – just:

- a. **Read** self-check #28 and #30 on page 117.
- b. **Think** about them a bit (so that you have a solution or part of a solution in mind).
- c. **Check** your thoughts by reading the answers at the end of the Chapter (on page 153).