Quiz 5 Writing even more Python functions Name _____

For the following questions, write your code in the space provided, or circle the letter of the MOST correct answer.

1. Write a function called count_cat that receives one string parameter. It counts the number of times the letters "cat" appear in the string, and returns that number.

```
Example calls:
a = count_cat("hellocatworld")
b = count_cat("cat hello cat world cat")
c = count_cat("hello world")
print a, b, c

Prints:
1 3 0
```

1.

2. Write a function named unlucky_7 that receives 2 numeric parameters. It calculates the sum of both numbers, except for numbers that are 7. They are skipped. It returns the resulting sum.

```
Example calls:
a = unlucky_7(3, 5)
b = unlucky_7(3, 7)
c = unlucky_7(7, 5)
d = unlucky_7(7, 7)
print a, b, c, d

Prints:
8 3 5 0
```

1.

3. Write a function named big_sum that receives 1 parameter, a list of numbers. It returns the sum of the largest value and the smallest value in the list. The list will always have at least 2 numbers.

```
Example calls:
a = big_sum([ 1, 2, 3, 4, 5 ])
b = big_sum([ 1, 5, 3, 4, 2 ])
print a, b

Prints:
6 6
```

1.

4. Write a function named adder that receives no parameters. It prompts the user for numbers until the user enters 0. It adds up all of the numbers typed by the user. It returns the total.

```
Example call:
a = adder()
print a

Output and user input:
Number? 5
Number? 6
Number? 0
11
```

1.

5. The function line_count already exists, as show below. It receives the name of a file as a string and returns the number of lines in the file.

   Write calls to this function to count the number of lines in the files poe.txt and asimov.txt, then display the number of lines in each.

Provided code:

```
def line_count(filename):
    count = 0
    f = open(filename, "rb")
    for line in f:
        count += 1
    f.close()
    return count
```

1.

6. Select the box that shows the output after the program is executed.

```
def problem6(s):
    r = ""
    for c in s:
        r = c + r
    return r

a = problem6("cat")
print a
```

1.

(a) cat	(b) tca	(c) tac	(d) atc

7. Select the box that shows the output after the program is executed.

```
def problem7(lst):
    for w in lst:
        if w == "dog":
            return True
    return False

a = problem7(["cat", "hamster", "snake", "worm"])
b = problem7(["cat", "hamster", "dog", "worm"])
print a, b
```

2.

(a) False False	(b) False True	(c) True False	(d) True True

8. Select the box that shows the output after the program is executed.

```
def problem8(day, insession):
    if (not insession) or (day < 1 or day > 5):
        return "none"
    elif day == 1 or day == 3 or day == 5:
        return "10:00"
    elif day == 2:
        return "9:00"
    elif day == 4:
        return "11:00"
    else:
        return "none"

a = problem8(2, False)
b = problem8(4, True)
c = problem8(6, True)
print a, b, c
```

1.

(a) none 11:00 none	(b) 9:00 none none	(c) 9:00 10:00 11:00	(d) 9:00 11:00 none

9. Select the box that shows the output after the program is executed.

```
def problem9(s):
    t = ""
    for i in range(len(s)):
        if i % 3 == 0:
            t += s[i]
    return t

a = problem9("39817645")
print a
```

1.

(a) 3874	(b) 39165	(c) 314	(d) 398

10. Select the box that shows the output after the program is executed.

```
def problem10(s):
    t = ""
    for i in range(len(s)):
        if s[i] == "w" and i < len(s) - 1:
            t += s[i+1]
    return t

a = problem10("what's the owl say? who!")
print a
```

1.

(a) hlh	(b) www	(c) 0 11 19	(d) 30