SOPH 303 Introduction to the Digital World: Final Practice

1. Write a function, `hello(name)`, that takes an input string and returns a salutation: ‘‘Hello, name. Your name contains n letters.’’, where n represents the number of letters in name.

   1. Test case 1: name = ‘‘John’’; hello(name) = ‘‘Hello, John. Your name has 4 letters.’’
   2. Test case 2: name = ‘‘Marina’’; hello(name) = ‘‘Hello, Marina. Your name has 6 letters.’’

2. Write a function, `f(x)`, that takes a string input x and returns the largest integer that is no greater than x. If x is not an integer or floating point number, return 0.

   1. Test case 1: x = ‘5.5’, f(x) = 5.
   2. Test case 2: x = ‘−2’, f(x) = −2.
   3. Test case 3: x = ‘a’, f(x) = 0.

3. Write a function, `mirror(t)`, that takes a tuple and returns a new tuple which is the input tuple concatenated with the reverse of itself.

   1. Test case 1: t = (1, 2, 3), mirror(t) = (1, 2, 3, 3, 2, 1).
   2. Test case 2: t = (1,), mirror(t) = (1, 1).
   3. Test case 3: t = (), mirror(t) = ().

4. Write a function, `count3(l)`, that takes a list l and returns the count of all positive numbers that are a multiple of 3 in the input list l.

   1. Test case 1: l = [1, 2, 3, 3, 3, 7], count3(l) = 3.
   2. Test case 2: l = [−3, 0], count3(l) = 0.
   3. Test case 3: l = [12], count3(l) = 1.

5. Write a function, `tcid(d)`, that takes an input dict, and returns a new dict whose keys are the values of the input dict, and whose values are the numbers of keys in the input dict that maps to the value.

   1. Test case 1: d = {1: ‘a’, 2: ‘b’}, tcid(d) = {‘a’: 1, ‘b’: 1}.
   2. Test case 2: d = {1: 5, 2: 6, 3: 5, 4: 0}, tcid(d) = {0: 1, 5: 2, 6: 1}.
6. Write a function, `mapeven(f, l)`, that takes an input function `f` and an input list `l`, and returns a list that contains `f(x)` for every second `x` in `l`.

   1. Test case 1: `f = abs, l = [-1, -2, -3], mapeven(f, l) = [2].`
   2. Test case 2: `f = lambda x: x + 1, l = [1, 3, 5, 0], mapeven(f, l) = [4, 1].`
   3. Test case 3: `f = str, l = [1], mapeven(f, l) = [].

7. Consider the following program

   ```python
def thing(inputList):
    output = []
    y = 0
    i = 0
    while i < len(inputList):
        if y < 100:
            y = y + inputList[i]
            output.append(y)
        else:
            output.append(y)
        i = i + 1
    return output
```

   1. What is the value of `thing([1, 2, 3, 100, 4, 9, 500, 51, -2, 57, 103, 1, 1, 1, 1, -10, 207, 3, 1])`

   2. Write a single state machine class `MySM` such that `MySM().transduce(inputList)` gives the same result as `thing(inputList)`, if `inputList` is a list of numbers.

8. For each difference equation below, say whether, for a unit sample input signal: the output of the system it describes will diverge or not, and the output of the system it describes (a) will always be positive, (b) will alternate between positive and negative, or (c) will have a different pattern of oscillation.

   1. `10y[n] - y[n - 1] = 8x[n - 3]`
   2. `y[n] = -y[n - 1] - 10y[n - 2] + x[n]`
   3. `y[n] = -0.6y[n - 1] + 0.16y[n - 2] - 0.1x[n - 1]`