

## Final Exam Practice

### Classes:

1. Define a class called IceCream to help Baskin-Robbins manage their ice cream orders. An IceCream object represents a cone with varying flavors and number of scoops. An IceCream object starts out with no scoop but more can be added.

The IceCream class should have the following methods.

Method	Description
<code>__init__(self)</code>	Creates an IceCream and sets up any fields necessary.
<code>add_scoop(self, flavor, num_of_scoops)</code>	Adds the corresponding flavor and number of scoops to IceCream.
<code>get_flavor(self, flavor)</code>	Returns the number of scoops of the given flavor in this IceCream. Returns zero if the flavor is not on the cone.
<code>to_string(self)</code>	Returns a string representation of the IceCream in the form: “<total number of scoops> scoops of ice cream with <list of flavors>”

## String Manipulation (Function Writing):

2. Write a function that takes a string and returns the same string but with even indices uppercased and odd indices lowercased.

Ex: `swap_casing('hello world')` should return `'HeLIO WoRID'`

## Dictionaries (Function Writing):

3. Write a function that takes in a dictionary mapping integers to strings. If the key is divisible by two, set the value equal to "even". Return a list containing the original values of the even keys.

Ex: `even_key({2:"hello", 3:"one", 4: "cat"})` should return `["hello", "cat"]` and the dictionary modified to look like this `{2:"even", 3:"one", 4: "even"}` after the function call.

## Debugging:

4. You receive the following error messages after running your code:

```
Traceback (most recent call last):
File "social_network.py", line 338, in <module>
do_stuff(friends_list)
File "social_network.py", line 200, in do_stuff
result = recommend_by_influence(friendlist)
File "social_network.py", line 107, in
recommend_by_influence
output = read_result()
NameError: global name 'read_result' is not defined
```

- a. List the names of the stack frames that existed at the point that the error was discovered?
- b. What is the most recent stack frame (eg. the last function that was successfully called)?
- c. Describe how you would go about trying to find the cause of and fix the error:

## Function Output:

5. Write the output of the code below:

```
sum = 0
for x in range(1, 25, 2):
    temp = (x / 10) % 10
    sum = sum + temp
print('sum:', sum)
```

## Function Documentation:

6. Write a docstring for the following function. Document the inputs, outputs, and any side effects of the function.

```
def unknown_func(num_list):
    unique_nums = set(num_list)
    print("There are", len(unique_nums), "unique numbers in the list")
    factors = {}
    for num in unique_nums:
        factors[num] = set()
        for factor in range(1, num):
            if num % factor == 0:
                factors[num].add(factor)
    return factors
```

## List Comprehension:

7. You are given the following class:

```
class Dog:
    def __init__(self, name):
        self.name = name

    def get_name(self):
        return self.name
```

In your main function, you have a list of dog names:

```
dog_names = ['Spot', 'Woofy', 'Fluffy', '']
```

Write a list comprehension to create a list of Dog objects, one with each name in the list of dog\_names, excluding names that are empty strings.