

## CSE 160 Section 7 Problems

1. For each line of code, determine if the code is reassigning a variable or mutating an object.

Code	Reassigning/Mutating
<code>my_variable = 2</code>	
<code>favorite_characters = ['Aether', 'Karma', 'Ukyo']</code>	
<code>least_favorite = favorite_characters</code>	
<code>best_friends.append('Ash')</code>	
<code>lst1 = mystery1()</code>	
<code>lst1[2] = 'Xiao'</code>	
<code>classes = {'cse160': 'Data Programming', 'cse154': 'Web Programming'}</code>	
<code>favorite_numbers['Chao'] = 43</code>	

2. Given the following code, figure out what would be printed out at the numbered lines.

```
students = ['Jakob', 'Marty', 'Karl', 'Xander']
class2 = students
print(students is class2) # 1
print(students == class2) # 2
class2.append('Joe')
print(students == class2) # 3
print(students[-1]) # 4
class3 = class2[:]
print(class2 is class3) # 5
print(class2[:] == class3[:]) # 6
sorted_students = sorted(students)
print(students == sorted_students) # 7
print(class2 is sorted_students) # 8
students.sort()
print(class2 is students) # 9
```

3.

a. Given the following functions and code, figure out what would be printed.

```
def increment_dict(dict, key):
    dict[key] = dict[key] + 1

def increment_list(list, idx):
    list[idx] = list[idx] + 1

def increment_number(number):
    number = number + 1

coins = {'Sam': 1}
increment_dict(coins, 'Sam')
print(coins)

numbers = [1, 2, 3]
increment_list(numbers, 1)
print(numbers)

my_num = 9
increment_number(my_num)
print(my_num)
```

b. The intent of these three functions is to increment the value by one. If any of the examples did not do this successfully, change the code such that it actually works.

4. Consider the following function `average`, which when given a list of numbers computes the average. For example, `average([1, 2, 3, 4, 5])` would return 3.0

```
def average(nums):
    total = 0
    for n in nums:
        total = total + n
    return total / len(nums)
```

a. Are there any potential problems with this function? What exceptions could be raised, if any?

i. Hint: what happens if you call `average([])`?

b. If there are any exceptions, how could you fix the function to avoid this problem? There may be multiple ways to do this.

5. Write code that will successfully modify the given variable, a list of tuples. After your code executes, each tuple in the list should have the field at index 1 incremented by 1.

```
likes = [('Chewbarka', 163), ('Mary Puppins', 15),
         ('Barkley', 4)]
```

6. Write a function `favorite_number` that is given a dictionary of names mapped to their favorite number, and a name. The function should return the favorite number of the given name. If the name is not in the dictionary, a `ValueError` should be raised. For example:

```
people = {
    'Chongyun': 7,
    'Razor': 1,
    'Amber': 4
}
print(favorite_number(people, 'Amber')) # 4
print(favorite_number(people, 'Xiao')) # ValueError
```

7. You are given a function `do_cool_thing()` that will *usually* do a cool thing, but randomly will throw an exception. Write code that calls this function, and in the event of an exception, prints some text to the console instead of crashing.
8. Write a function `least_coins` that takes a dictionary as a parameter, and returns the name of the person with the fewest coins. For example, the following code should print “Alex”:

```
coin_totals = {
    'Sam': 12,
    'Ash': 68,
    'Alex': 5,
    'Erin': 10
}
print(least_coins(coin_totals))
```

9. The social media company Yipper has asked you to help debug some code they've been having issues with. They've given you the following exception:

```
Traceback (most recent call last):
  File "yipper.py", line 21, in <module>
    likes = likes_per_user(yips)
  File "yipper.py", line 14, in likes_per_user
    result[i][1] = result[i][1] + yip['likes']
TypeError: 'tuple' object does not support item assignment
```

- What does this exception mean? What has gone wrong?
- The company has given you the following code. How could it be fixed to avoid this error?

```
def get_yips():
    return [
        {'name': 'Chewbarka', 'likes': 16,
         'message': 'Met a snake yesterday; seemed pretty cool \
                    #python'},
        {'name': 'Mary Puppins', 'likes': 0,
         'message': 'Today was pretty exceptional! \
                    #notavaluerror'},
        {'name': 'Chewbarka', 'likes': 3,
         'message': 'Started playing a new game today! #anemo'}
    ]

def likes_per_user(yips):
    result = []
    for yip in yips:
        found = False
        for i in range(len(result)):
            if result[i][0] == yip['name']:
                result[i][1] = result[i][1] + yip['likes']
                found = True
        if not found:
            result.append((yip['name'], yip['likes']))
    return result

yips = get_yips()
likes = likes_per_user(yips)
print(likes)
```