“Homework12” Justin Minsk

1.

def find\_dubs(list):  
 count\_list = {} # create a dict count list  
 final\_list = [] # create a list named final list  
 for entry in list:  
 if entry in count\_list: # if entry is in count list  
 count\_list[entry].append(entry) # add the entry to the list side of the dict  
 else:  
 count\_list[entry] = [entry] # make the key the entry and add the entry to the list  
 for entry in count\_list:  
 if len(count\_list[entry]) >= 2: # if the list length is longer than or equal to 2  
 final\_list.append(entry) # add the entry to the list  
 print(final\_list) # print the final list  
 return final\_list # return the final list  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 intlist = [1, 2, 3, 3, 4, 4, 5, 6, 7, 3, 2, 3, 4, 3, 4, 5, 5]  
 find\_dubs(intlist)

2.

def mating\_pairs(males, females):  
 pairs = set() # create a empty set named pairs  
 while males.difference(females) != set(): # while either list is not empty  
 hold = males.pop() # save males pop  
 hold2 = females.pop() # save female pop  
 pairs.add((hold, hold2),) # add then into pairs as a set  
 print(pairs) # print pairs  
 return pairs # return pairs  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 mating\_males = {'male1', 'male2', 'male3'}  
 mating\_females = {'female1', 'female2', 'female3'}  
 mating\_pairs(mating\_males, mating\_females)  
 mating\_males = {1, 2, 3}  
 mating\_females = {11, 22, 33}  
 mating\_pairs(mating\_males, mating\_females)

3.

def count\_values(dict):  
 count = 0 # create a count  
 for entry in dict:  
 if dict[entry] == 1: # if the number value is equal to 1  
 count += 1 # increase count by 1  
 print(count) # print count  
 return count # return count  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 dict = {'red': 1, 'green': 1, 'blue': 2}  
 count\_values(dict)

4.

def count\_dup(dict):  
 count = 0 # create a count  
 for entry in dict:  
 if dict[entry] > 1: # if the the list for each key of the dict is greater than 1  
 count += 1 # increase count by one  
 print(count) # print count  
 return count # return count  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 dict = {'red': 1, 'green': 1, 'blue': 2}  
 count\_dup(dict)

5.

def dict\_interest(dict1, dict2):  
 end\_result = {} # create the dict we will enter into  
 for (index, value) in dict1.items(): # look at both key and value in dict1  
 if index in dict2 and value == dict2[index]: # if key is in dict2 and value equals value in dict2  
 end\_result[index] = value # add the key and value to the end result  
 print(end\_result) # print end result  
 return end\_result # return end result  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 dict1 = {'hi': 1, 'bye': 2, "i": 3}  
 dict2 = {'hi': 2, 'bye': 2, "j": 3}  
 dict\_interest(dict1, dict2)