```
# 1.

def common (t1,t2):
    """ Assumes t1&t1 are tuples.
        Returns tuple result with their common terms. """
    result=()
    for i in t1:
        if i in t2 and i not in result:
            result=result+(i,)
    return result

# 2.

def factor (x,y):
    """ Assumes x&y are integers.
        Returns a tuple with their least and greatest common factor."""
    common=()
    for i in range (2,abs(x)+1):
        if x%i==0 and y%i==0:
            common=common+(i,)
    ans=((min(common),max(common)))
    return ans
# 3.

def single_copy (L):
    """ Assumes L is a list, the function delete duplicates from
         L."""
    L1=[]
    L2=L[:]
    for i in L2:
        L.remove(i)
        if i not in L1:
           L.append(i)
        L1.append(i)

#check:
#y=[45,"op",99,45,"op",78,105]
#single_copy(y)
#print (y)

# 4.

def count_terms_in_interval (L,x,y):
    """Assuming L is a list of floats, x&y are floats.
       Returns the number of times z appear on the list where
       min(x,y)<z<max(x,y)"""
    num=0
    for i in L:
        if min(x,y)<i and i<max(x,y):
            num=num+1
    return num

# check:
P=[23,56.67,-23,90,78,0,900,23]
print (count_terms_in_interval(P,2,100))

# 5.
def common3(L1,L2,L3):
    """ Assuming L1,L2&L3 are lists.
        Changes L1 and leaves in it only the terms that are common to the 3 original lists.
        The function does not remove duplicates """
    L1_copy=L1[:]
    for i in L1_copy:
        if i not in L2 or i not in L3:
             L1.remove(i)
# check:

R1=[34,67,67,78,99,12,123]
R2=[67,12,123,34]
R3=[67,123]
common3(R1,R2,R3)
print(R1)
```