In this version, the value of s is 'microwaves'

problem 1

---------

a) 'o'

b) 4

c) [15]

d) 'owave'

e) [15, 4, 0]

problem 2

---------

a) [0, 3, 6, 9]

b) ['t', 'h']

c) [2, 2, 2, 3, 5]

d) [False, False, True, False, True]

problem 3

---------

a) mystery(3, 7) calls mystery(4, 5)

mystery(4, 5) calls mystery(5, 3)

mystery(5, 3) returns 3

mystery(4, 5) returns 4 + 3 = 7

mystery(3, 7) returns 3 + 7 = 10

b) 10

problem 4

---------

def remove\_spaces(s):

if s == '':

return ''

else:

rem\_rest = remove\_spaces(s[1:])

if s[0] == ' ':

return rem\_rest

else:

return s[0] + rem\_rest

problem 5

---------

foo 3 5

foo 8 6

6 5

foo 5 5

foo 10 8

6 5

problem 6

---------

# solution using a list comprehension

def min\_pos(values):

pos\_vals = [x for x in values if x > 0]

return min(pos\_vals)

# solution using a recursive helper function

def pos\_values(values):

if values == []:

return []

else:

pos\_rest = pos\_values(values[1:])

if values[0] > 0:

return [values[0]] + pos\_rest

else:

return pos\_rest

def min\_pos(values):

pos\_vals = pos\_values(values)

return min(pos\_vals)

problem 7

---------

def num\_occur(val, values):

if val not in values: # values == [] would also work

return 0

else:

num\_in\_rest = num\_occur(val, values[1:])

if values[0] == val:

return 1 + num\_in\_rest

else:

return num\_in\_rest

problem 8

---------

fad

fit

fee