### Problem 1

0.0/10.0 points (graded)

Assume s is a string of lower case characters.

Write a program that counts up the number of vowels contained in the string s. Valid vowels are: 'a', 'e', 'i', 'o', and 'u'. For example, if s = 'azcbobobegghakl', your program should print:

Number of vowels: 5

s = 'azcbobobegghakl'

total = 0

for char in s:

    if char in "aeiou":

        total += 1

print(total)

### Problem 2

0.0/10.0 points (graded)

Assume s is a string of lower case characters.

Write a program that prints the number of times the string 'bob' occurs in s. For example, if s = 'azcbobobegghakl', then your program should print

Number of times bob occurs is: 2

s = 'azcbobobegghakl'

i = 0

count = 0

for char in s:

    if s[i:i + 3] == "bob":

        count += 1

    i += 1

print("Number of times bob occurs is:", count)

Problem 3

0.0/15.0 points (graded)

Assume s is a string of lower case characters.

Write a program that prints the longest substring of s in which the letters occur in alphabetical order. For example, if s = 'azcbobobegghakl', then your program should print

Longest substring in alphabetical order is: beggh

In the case of ties, print the first substring. For example, if s = 'abcbcd', then your program should print

Longest substring in alphabetical order is: abc

Note: This problem may be challenging. We encourage you to work smart. If you've spent more than a few hours on this problem, we suggest that you move on to a different part of the course. If you have time, come back to this problem after you've had a break and cleared your head.

s = 'zyxwvutsrqponmlkjihgfedcba'

longeststring = s[0:1]

# Iterate over index and string

for char in range(0, len(s)-1):

    start = char

    length = 0

    while s[start + length] <= s[start + length + 1]:

        length += 1

        if len(longeststring) < length + 1:

#            print(s[char], s[start + length],

#                  s[start: start + length + 1], length + 1)

            longeststring = s[start:start + length + 1]

        if start + length + 1 == len(s):

            break

print("Longest substring in alphabetical order is:", longeststring)

Version 2

s = 'zyxwvutsrqponmlkjihgfedcba'

longeststring = s[0:1]

# Iterate over index and string

for char in range(0, len(s)-1):

    start = char

    length = 1

    while s[start + length - 1] <= s[start + length]:

        length += 1

        if len(longeststring) < length:

            longeststring = s[start:start + length]

        if start + length == len(s):

            break

print("Longest substring in alphabetical order is:", longeststring)

Version 3

s = 'zyxwvutsrqponmlkjihgfedcba'

tmpString = longestString = s[0:1]

for i, char in enumerate(s):

    if char <= s[i + 1:i + 2]:

        tmpString += s[i+1:i + 2]

        if len(longestString) < len(tmpString):

            longestString = tmpString

    else:

        tmpString = longeString = s[i+1:i + 2]

print("Longest substring in alphabetical order is:", longestString)