

## Assessment Problem Definition – Part 2:

Use whatever programming language you are most comfortable with to program a solution for the problems presented below.

Please take 30 minutes to complete the assessment. After the 30 minutes is complete, please zip your solution and return the solution for evaluation.

**[Problem 1]** In this problem we are asking for implementation of a function, *f*, that encodes a String.

The function *f* shall accept a String as an input and return the encoded String as a return value.

The encoding algorithm replaces every character in the string with the letter that has exactly the same position when counting backwards from letter “z” (with alphabet sort order in mind.)

Non-alphabet characters are ignored.

Here are three examples:

[a] Letter “a” is replaced by letter “z” (because “a” is the first letter of alphabet and “z” is the first letter when counted from the end of the alphabet list.)

[b] Letter “c” is replaced by letter “x” (because “c” is the third letter of alphabet and “x” is the third letter of alphabet when counted backwards from letter “z”.)

[c] Letter “p” is replaced by letter “k” (because “p” is the 16th letter of alphabet and “k” is the 16th letter when counted backwards from letter “z”.)

Therefore, when string “acp” is encoded, the return value is string “zxk”.

Please write and submit this function with any unit tests that you would like to include.

What would be the result of encoding the String “Errors in strategy cannot be correct through tactical maneuvers”?

**[Problem 2]** In this problem we are asking for implementation of a function, *g*, that returns the frequency of alphabet letters in a given string ignoring upper/lower case issues.

Use whatever programming language you are most comfortable with, although we prefer to see standard scripting languages used.

The return format is best demonstrated in an example.

Suppose you are given the String “Hello there! Apple!”

The function *g* shall return the following structure:

```
{  
    a:1,  
    b:0,  
    c:0,  
    d:0,  
    e:4,  
    f:0,  
    g:0,  
    h:2,  
    i:0,  
    j:0,  
    k:0,  
    l:3,  
    m:0,  
    n:0,  
    o:1,  
    p:2,  
    q:0,  
    r:1,  
    s:0,  
    t:1,  
    u:0,  
    v:0,  
    w:0,
```

x:0,

y:0,

z:0

}