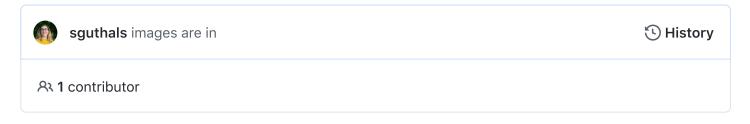


aed-wonder-woman / 0-wonderwomanlp / 2-secretmessage / includes / 4-lasso-letter-decode-function-part-2.md





# Decoding One Letter at a Time Part 2

Figuring out the **true** letter code that matches the decoded letter will involve cycling around the alphabet. So, if your letterCode + shiftAmount is equal to or greater than 123, you need to return back to 97 to continue counting.

How do you get 123? It's just 97 (the letter code for the 'a' character) plus 26 (the number of letters in the alphabet). And if you look at the ASCII number for 'z', you will see that it is 122!

So you could write conditional statements to check this, but there is an easier way!

### **The Mod Operator**

To wrap around the alphabet the easy way, you will need a special operator called mod and represented as %.

Mod (%) will divide two numbers and return the remainder. So, for example, if you ran the following code in Python:

```
threeTwo = 3 % 2
elevenFour = 11 % 4
fiveTen = 5 % 10
```

#### You would get:

Variable	Formula	Value
threeTwo	3/2 = 1 remainder 1	1
elevenFour	11/4 = 8 remainder 3	3
fiveTen	5/10 = 0 remainder 5	5

## Calculating the Decoded Character: The right way

With the mod operator in mind, you will need two new variables:

- aAscii: holds the ASCII value of 'a' by calling ord('a')
- alphabetSize: holds the number of letters in the alphabet, so 26!

Then, the formula for figuring out the trueLetterCode would be: aAscii + (((letterCode - aAscii) + shiftAmount)% alphabetSize)

You can review this formula with a couple examples:

### Example 1: 'a' and 2

Start with:

- letter = 'a'
- shiftAmount = 2

Variable	Formula	Value
letter	none	'a'
shiftAmount	none	2
letterCode	ord('a')	97
aAscii	ord('a')	97
alphabetSize	none	26

Variable	Formula	Value
trueLetterCode	97 + (((97 - 97) + 2) % 26)	2 (see below)
decodedLetter	chr(99)	С

You can review the formula for trueLetterCode just like you would any other math formula (remember PEMDAS): aAscii + (((letterCode - aAscii) + shiftAmount))% alphabetSize) 97 + (((97 - 97) + 2)) % 26) 97 + ((0 + 2)) % 26) 97 + (2) % 26) 97 + (2) % 29

### Example 2: 'W' and 13

Start with:

- letter = 'W'
- shiftAmount = 13

Variable	Formula	Value
letter	none	'W'
shiftAmount	none	13
letterCode	ord('w')	119
aAscii	ord('a')	97
alphabetSize	none	26
trueLetterCode	97 + (((119 - 97) + 13) % 26)	106 (see below)
decodedLetter	chr(106)	j

You can review the formula for trueLetterCode just like you would any other math formula (remember PEMDAS): aAscii + (((letterCode - aAscii) + shiftAmount) % alphabetSize) 97 + (((119 - 97) + 13) % 26) 97 + ((22 + 13) % 26) 97 + (35 % 26) 97 + 9 106

#### **Final Code**

Now that you have your decoder formula, you can put it all together in your function (make sure you add comments so you can remember what is happening!):

# define a function that finds the truth by shifting the letter by the specif
def lassoLetter( letter, shiftAmount ):

```
# invoke the ord function to translate the letter to its ASCII code
# and save it to the variable called letterCode
letterCode = ord(letter.lower())

# the ASCII number representation of lowercase a
aAscii = ord('a')

# the number of letters in the alphabet
alphabetSize = 26

# the formula to calculate the ASCII number for the decoded letter
# taking into account looping around the alphabet
trueLetterCode = aAscii + (((letterCode - aAscii) + shiftAmount) % alphab

# converting the ASCII number to the character/letter
decodedLetter = chr(trueLetterCode)

# sending the decoded letter back
return decodedLetter
```

Now that you have the lassoLetter() function you can call this on each letter in the secret message!

WONDER WOMAN 1984 TM & © DC and WBEI. RATED PG-13