**Problem Set 5 - Strings**

**p1: rotateWord.py**

A Caesar cypher is a weak form of encryption that involves 'rotating' each letter by a fixed number of places. To rotate a letter means to shift it through the alphabet, wrapping around to the beginning if necessary, so 'A' rotated by 3 is 'D' and 'Z' rotated by 1 is 'A'.

To rotate a word, rotate each letter by the same amount. For example, 'cheer' rotated by 7 is 'jolly' and 'melon' rotated by -10 is 'cubed'. In the move *2001: A Space Odyssey*, the ship computer is called HAL, which is IBM rotated by -1.

Write a function called `rotate_word()` that takes a string and an integer as parameters, and returns a new string that contains the letters from the original string rotated by the given amount.

You might want to use the built-in function `ord()`, which converts a character to a numeric code, and `chr()`, which converts numeric codes to characters. Letters of the alphabet are encoded in alphabetical order, so for example:

```python
>>> ord('c') - ord('a)
2
```

Because 'c' is the two-eth letter of the alphabet. But beware: the numeric codes for uppercase letters are different.

Potentially offensive jokes on the Internet are sometimes encoded in ROT13, which is a Caesar cypher with rotation 13. If you are not easily offended, find and decode some of them.