### CMSC 143: Introduction to Object-Oriented Programming with Robots

# Lab 5: ROT13 and Recursion
# Due October 5, 2009

Submit a copy of your python program (cmsc143_lab5_LASTNAME_FIRSTNAME.py) on moodle. Your program should have your name, email, and the date at the top of the file as a comment.

## Learning Objectives

○ Work with strings. ○ More practice with loops. ○ Use recursion.

## ROT-13

From Chapter of 8 of Think Python: ROT13 is a weak form of encryption that involves 'rotating' each letter in a word by 13 places[1]. To rotate a letter means to shift it through the alphabet, wrapping around to the beginning if necessary, so 'A' shifted by 3 is 'D' and 'Z' shifted by 1 is 'A'. Potentially offensive jokes on the Internet are sometimes encoded in ROT13.

You might want to use the built-in functions `ord(ch)`, which converts a character to a numeric code, and `chr(i)`, which converts numeric codes to characters.

1. `rot13(msg)` Write a function called `rot13(msg)` that takes a string and encodes it in ROT13. The same function will decode messages, as well.

2. `rot13R(msg)` Rewrite the function called `rot13(msg)` so that it uses recursion (or `for` loops if you used recursion in `rot13`).

3. `rotMsg(msg, n)` Write a function called `rotMsg(msg, n)` 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.

   For example, "cheer" rotated by 7 is "jolly" and "melon" rotated by -10 is "cubed".

4. `getCodeName()` Write a function `getCodeName()` that returns your robot's name, encoded in ROT-13. What is the result?

[1] http://en.wikipedia.org/wiki/ROT13

## Drawing trees that draw trees that draw trees that draw trees ...

The following function uses your robot to draw trees. Well, almost.

1. The first problem is that the code doesn't work, there is a small bug.
2. Find two interesting sets of values for tree-parameters (size and angle). Write a short description of the graphics they create (better yet, include a screenshot of your creation). Include this description as a comment in your program.
3. The second problem is that this program uses the turtle package which allow you do create some simple graphics, but not the robot. Change the program to make your robot draw the tree.
4. Modify the turtle program to draw some other interesting shape (a more interesting tree, perhaps).

```
from turtle import *

def tree(size, angle):

    # move forward
    forward(size)

    # draw the left tree
    left(angle)
    tree(size/1.7, angle)

    #turn back toward the center
    right(angle)

    #draw the right tree
    right(angle)
    tree(size/1.6, angle)

    #turn back toward the center
    left(angle)

    #move back to where we started
    backward(size)

reset()
tree(100.0, 20)
```