Homework 5

Integrity: Your honor is extremely important. This academic security policy is designed to help you succeed in meeting academic requirements while practicing the honorable behavior our country rightfully demands of its military. Do not compromise your integrity by violating academic security or by taking unfair advantage of your classmates.

Authorized Resources: You can **ONLY** seek help from the instructor. Homework is an individual exercise.

Inverse Kinematics

Create a `Jupyter notebook` and with this at the top.

```
%matplotlib inline
```

```
from __future__ import division, print_function
from matplotlib import pylab
```

The robot arm is commanded by an ASCII string that looks like this: `#0 P1000 #1 P2000 T2500\r`. This string tells servo 0 to set a PWM of 1000 and servo 1 set to a PWM of 2000. The last part of the command T2500 basically gives a time frame to move the arm to the new position. If we move too fast, we could damage the arm, so we are always going to send the last command as T2000 since high speed is not important to us. Now this command only moves the first 2 servos, but our arm has 5 degrees of freedom (5 servos), so our command string will contain 5 servo commands and the T2500 command with a `\r` on the end.

This homework will walk you through building the code to run the robot arm. In the lab, you will need to calibrate your servos. For now, we will use these values which are close. Make sure you write your code so it is easy to change these values later.

Angle [deg]	PWM
0	900
180	2100

1. Write a function that takes in an angle (degrees or radians, your choice) and returns a string to command 1 servo to that position. There is a linear relationship (i.e., straight line) between angle and PWM. For radians, you should get `angle2pwm(2)` = 1663.

   ```
   def angle2pwm(angle):
       """
       returns pwm counts
       """
       ...
   ```

   Now try `angle2pwm(1)` = ???

2. Using the function above, write another function that takes in 5 angles and returns the ASCII command string. For radians, you should get `command(1,2,3,2,1)` = `#0 P1281 #1 P1663 #2 P2045 #3 P1663 #4 P1281 T2500\r` where `\r` is a return in ASCII.

   ```
   def command(a, b, c, d, e):
       """
       returns the servo controller string #0 ... T2500\r
       """
       ...
   ```

   Hint: Remember to append `T2500\r` on the end

   Now try command(3,2,1,2,3), what is your command string?

3. Write a function to calculate cosine law. For radians, you should get `cosine_law(5,5,5)` = $\pi/3$. This is an equilateral triangle where all angles are the same.

   ```
   def cosine_law(a, b, c):
       """
       Where a and b are the sides and c is opposite the angle you want to find.
       It should return angles in radians
       cosine_law(5,5,5) -> pi/3 or 60 degrees
       """
       ...
   ```

   Now try `cosine_law(2,5,4)` = ??? in degrees

4. Write a function that takes a 3d point (x, y, z) and returns the joint angles. Given `inverse(3,3,3,0,0)`, you should get: `(45, 182.6, 156.4, 26.3, 0)`

```
def inverse(x,y,z, orientation, claw):
    """
    Calculates the joint angles given:
       (x,y,z) - end effector location
       orientation - orientation of end effector
       claw - is the claw open or closed
    """
    ...
```

Now try `inverse(3,4,5, 90, 0)`