import serial

import time

import math

import os

import serial

def getFrontSensor(ser, sensorMat):

ser.reset\_input\_buffer()

sensor = ser.readline()

sensorMat = sensor.split(",")

print sensorMat[1]

sensor\_distance = int(sensorMat[1])

return sensor\_distance

def getLeftSensor(ser, sensorMat):

ser.reset\_input\_buffer()

sensor = ser.readline()

sensorMat = sensor.split(",")

print sensorMat[0]

sensor\_distance = int(sensorMat[0])

return sensor\_distance

def getRightSensor(ser, sensorMat):

ser.reset\_input\_buffer()

sensor = ser.readline()

sensorMat = sensor.split(",")

print sensorMat[2]

sensor\_distance = int(sensorMat[2])

return sensor\_distance

xpos\_robot = int(raw\_input("Robot X Position: "))

ypos\_robot = int(raw\_input("Robot Y Position: "))

xpos\_goal = int(raw\_input("Goal X Position: "))

ypos\_goal = int(raw\_input("Goal Y Position: "))

distance = math.sqrt((xpos\_goal - xpos\_robot)\*\*2 + (ypos\_goal - ypos\_robot)\*\*2)

angle = round(math.degrees(math.atan2((ypos\_goal - ypos\_robot), (xpos\_goal - xpos\_robot))))

print distance, angle

ser = serial.Serial('/dev/ttyUSB0', 115200)

not\_at\_goal = True

distance\_traveled = 0

sensorMat = [0, 0, 0]

os.system('/root/rc\_wheeled\_auto/rc\_wheeled\_auto ' + str(angle) + ' ' + str(0))

while not\_at\_goal:

barrier = getFrontSensor(ser, sensorMat)

if (barrier < 10):

print "Barrier"

os.system('/root/rc\_wheeled\_auto/rc\_wheeled\_auto ' + str(90) + ' ' + str(0))

while(barrier > 0 and barrier < 15):

os.system('/root/rc\_wheeled\_auto/rc\_wheeled\_auto ' + str(0) + ' ' + str(1))

barrier = getLeftSensor(ser,sensorMat)

os.system('/root/rc\_wheeled\_auto/rc\_wheeled\_auto ' + str(0) + ' ' + str(1))

os.system('/root/rc\_wheeled\_auto/rc\_wheeled\_auto ' + str(-90) + ' ' + str(1))

else:

os.system('/root/rc\_wheeled\_auto/rc\_wheeled\_auto ' + str(0) + ' ' + str(1))

distance\_traveled += 1

if (distance - distance\_traveled) < 0:

not\_at\_goal = False;

ser.close()