```
/*

Gesture recognition code

This is a python code save it as mit.py and compile it

*/




#!/usr/bin/python

import serial

import time

import os

#import SendKeys

#import win32com.client


#shell = win32com.client.Dispatch("Wscript.Shell")

ser = serial.Serial('/dev/ttyACM0', 9600)

ser.open()

def rest():

        print "null\n"

def app1():

        print "success1\n"

def app2():

        print "success2\n"


def app3():

        print "success3\n"


def app4():

        print "success4\n"
```

```python
def fun1():
    flag=0
    start1 = time.time()
    elapsed = time.time() - start1
    while (1):
        threshold_time = 0.5
        elapsed = time.time() - start1
        try:
            #time.sleep(0.5)

            result = ser.readline()
        # print 'result',result

            a,b,c,d = result.split(" ")
            values = {'a':int(a),'b':int(b),'c':int(c),'d':int(d)}
            #print values
        except:
            continue



        if flag==0:
            for (sensor,value) in values.iteritems():
                if int(value) > 200:
                    flag=1
                    s=sensor
                    #print sensor
                    break
        #elif flag==0 and elapsed > 0.02:return '0'

        else:
```

```python
                for (sensor1,value1) in values.iteritems():
                    if sensor1==s and value1 < 150:
                        #start1 =
                        return sensor
                        flag=0
                        break
            if flag==0:
                elapsed = time.time() - start1
                if elapsed > threshold_time:
                    break


        #if elapsed > 0.01:
            #return '0'
        return '0'
try:
        path = []
    while 1:
                values = []
                a = fun1()
                #print a, path
                if a == '0':
                    if len(path) > 0:
                        print path
                        #print "1"
                        #if path == ['a']:


                            #os.system('libreoffice --view 1.odp')
                    if path == ['a','b','c']:


                            os.system('xdotool search "LibreOffice Impress" windowactivate
--sync key F5')
```

```python
        elif path == ['a','b']:
            os.system('xdotool key Down')
                    #time.sleep(1)
        elif path == ['b','a']:
                os.system('xdotool key Up')
                    #time.sleep(1)
        elif path == ['b','c']:
                        os.system('xdotool search "Flash" windowactivate --
sync key ctrl+Return')


        elif path == ['b']:
                os.system('xdotool key space')


        elif path == ['c','b']:
                        os.system('xdotool search "VLC media player"
windowactivate --sync key space')


        #elif path == ['b','c']:
                #os.system('xdotool key ctrl+Up')


        #elif path == ['c','b']:
                #os.system('xdotool key ctrl+Down')


        elif path == ['a','d']:
                os.system('xdotool key p')


        elif path == ['d','a']:
                os.system('xdotool key n')


        elif path == ['d','c']:
```

```python
                    os.system('xdotool key ctrl+Right')
                elif path == ['c','d']:
                    os.system('xdotool key ctrl+left')


                path = []


            else:
                #do nothing
                path = []
                pass
        else:
            path.append(a)
            #print path


except KeyboardInterrupt:


    ser.close()
```