-

.

from Myro import \*

init("/dev/tty.Fluke2-09B7-Fluke2")

def drawA():

forward(.5, 1)

turnBy(60)

forward(.5, 3)

turnBy(-120)

forward(.5, 1.5)

turnBy(-120)

forward(.5, 1.5)

backward(.5, 1.52)

turnBy(120)

forward(.5, 1.5)

turnBy(60)

def drawB():

forward(.5, 1)

turnBy(90)

forward(.5, 3)

#arc1

turnBy(-100)

forward(.5, 1)

turnBy(-50)

forward(.5, .75)

turnBy(-60)

forward(.5, .75)

turnBy(-50)

forward(.5, 1.1)

#arc2

turnBy(170)

forward(.5, 1)

turnBy(-50)

forward(.5, .75)

turnBy(-60)

forward(.5, .75)

turnBy(-60)

forward(.5, 1.3)

#end

turnBy(180)

forward(.5, 2)

def drawC():

#first go

forward(.5, 2)

turnBy(170)

forward(.5, 1)

turnBy(-40)

forward(.5, 1.3)

turnBy(-40)

forward(.5, 1.3)

turnBy(-40)

forward(.5, 1)

turnBy(-40)

forward(.5, 1)

turnBy(-40)

forward(.5, 1)

#back it up

turnBy(180)

forward(.5, 1)

turnBy(40)

forward(.5, 1)

turnBy(40)

forward(.5, 1)

turnBy(40)

forward(.5, 1.3)

turnBy(40)

forward(.5, 1.3)

turnBy(40)

forward(.5, 1)

#base of the C

turnBy(30)

forward(.5, 1)

backward(.5, .8)

#end

turnBy(-10)

forward(.5, 1)

def drawD():

forward(.5, 1)

turnBy(90)

forward(.5, 3)

#arc

turnBy(-100)

forward(.5, 1)

turnBy(-30)

forward(.5, 1.2)

turnBy(-50)

forward(.5, 1.4)

turnBy(-50)

forward(.5, 1.2)

turnBy(-40)

forward(.5, 1.5)

#end

turnBy(180)

forward(.5, 2.3)

def drawE():

forward(0.5,0.2)

turnBy(90)

forward(0.5,3)

turnBy(-90)

forward(0.5,1.5)

backward(0.5,1.5)

turnBy(90)

backward(0.5,1.5)

turnBy(-90)

forward(0.5,1.5)

backward(0.5,1.5)

turnBy(90)

backward(0.5,1.5)

turnBy(-90)

forward(0.5, 2)

def drawF():

forward(0.5,0.2)

turnBy(90)

forward(0.5,3)

turnBy(-90)

forward(0.5,1.5)

backward(0.5,1.5)

turnBy(90)

backward(0.5,1.5)

turnBy(-90)

forward(0.5,1.5)

backward(0.5,1.5)

turnBy(90)

backward(0.5,1.5)

turnBy(-90)

forward(0.5, 2)

def drawG():

forward(0.5,2)

turnBy(90)

forward(0.5,1.3)

turnBy(90)

forward(0.5,1.3)

backward(0.5,1.3)

turnBy(-90)

backward(0.5,1.3)

turnBy(90)

forward(0.5,2)

turnBy(135)

backward(0.5,1)

turnBy(-45)

backward(0.5,2)

turnBy(135)

forward(0.5,1)

turnBy(135)

backward(0.5,2)

forward(0.5,2)

turnBy(-135)

backward(0.5,1)

turnBy(45)

backward(0.5,2)

turnBy(45)

backward(0.5,1)

turnBy(45)

backward(0.5,2)

turnBy(180)

forward(0.5,0.3)

def drawH():

forward(0.5,0.2)

turnBy(90)

forward(0.5, 3)

backward(0.5, 1.5)

turnBy(-90)

forward(0.5, 1.5)

turnBy(90)

forward(0.5, 1.5)

backward(0.5, 3)

turnBy(-90)

forward(0.5,0.3)

def drawI():

forward(0.5, 1.1)

turnBy(90)

forward(0.5,3)

turnBy(90)

forward(0.5,0.75)

backward(0.5,1.5)

forward(0.5,0.75)

turnBy(90)

forward(0.5,3)

turnBy(90)

forward(0.5,1.1)

def drawJ():

forward(.5, 1)

turnBy(-30)

backward(.5,.75)

turnBy(-30)

backward(.5, .75)

forward(.5, .75)

turnBy(30)

forward(.5, .75)

turnBy(60)

forward(.5, .75)

turnBy(60)

forward(.5, 2.7)

backward(.5, 2.7)

turnBy(-60)

backward(.5, .75)

turnBy(-30)

forward(.5, 1.3)

def drawK():

turnBy(90)

forward(.5, 3)

backward(.5, 1.5)

turnBy(-60)

forward(.5, 2)

backward(.5, 2)

turnBy(-60)

forward(.5, 2)

backward(.5, 2)

turnBy(-60)

forward(.5, 1.5)

turnBy(90)

forward(.5, 2)

def drawL():

turnBy(90)

forward(.5, 3)

backward(.5, 3)

turnBy(-90)

forward(.5, 1.5)

backward(.5, 1.5)

forward(.5,2.2)

def drawM():

turnBy(90)

forward(.5,3)

turnBy(-140)

forward(.5,2)

turnBy(100)

forward(.5,2)

turnBy(-140)

forward(.5,3)

turnBy(90)

forward(.5, .5)

def drawN():

turnBy(90)

forward(.5,3)

turnBy(25)

backward(.5,3.1)

turnBy(-25)

forward(.5,3.1)

backward(.5,3)

turnBy(-90)

forward(.5, 1)

def drawO():

forward(0.5,1)

motors(0.1,0.5,13.5)

forward(0.5,1.3)

def drawP():

forward(.5,.5)

turnBy(90)

forward(.5,3)

turnBy(-100)

forward(.5, 1)

turnBy(-50)

forward(.5, .75)

turnBy(-60)

forward(.5, .75)

turnBy(-50)

forward(.5,.9)

turnBy(80)

forward(.5,1.5)

turnBy(90)

forward(.5,2)

def drawQ():

forward(0.5,1)

motors(0.1,0.5,13.5)

forward(.5, .5)

turnBy(-45)

backward(.5, 1)

forward(.5, 1)

turnBy(45)

forward(0.5,1.3)

def drawR():

forward(.5, 1)

turnBy(90)

forward(.5, 3)

#arc1

turnBy(-100)

forward(.5, 1)

turnBy(-50)

forward(.5, .75)

turnBy(-60)

forward(.5, .75)

turnBy(-50)

forward(.5, 1.1)

turnBy(-45)

backward(0.5, 2.12)

turnBy(-135)

forward(.5,.3)

def drawS():

from Myro import \*

turnBy(90)

forward(.5,1)

backward(.5,1)

turnBy(-90)

forward(.5,1.5)

turnBy(90)

forward(.5,1.5)

turnBy(90)

forward(.5,1)

turnBy(-90)

forward(.5,1.5)

turnBy(-90)

forward(.5,1.5)

turnBy(-90)

forward(.5,1)

turnBy(180)

forward(.5,1)

turnBy(90)

forward(.5,1.5)

turnBy(90)

forward(.5,1.5)

turnBy(90)

forward(.5,1)

turnBy(-90)

forward(.5,1.5)

turnBy(90)

forward(.5,.3)

def drawT():

forward(0.5, 1.1)

turnBy(90)

forward(0.5, 3)

turnBy(90)

forward(0.5, 1.3)

backward(0.5,2.6)

forward(0.5, 1.3)

turnBy(90)

forward(0.5,3)

turnBy(90)

forward(0.5,1.1)

def drawU():

forward(0.5, 0.3)

turnBy(90)

forward(0.5, 3)

backward(0.5, 3)

turnBy(-90)

forward(0.5, 2)

turnBy(90)

forward(0.5, 3)

backward(0.5, 3)

turnBy(-90)

forward(0.5, 0.2)

def drawV():

from Myro import \*

##code here

forward(.5, 1.5)

turnBy(120)

forward(.5,2.5)

backward(.5,2.5)

turnBy(-60)

forward(.5,2.6)

backward(.5,2.5)

turnBy(-60)

forward(.5,1)

def drawW():

forward(.5, 2)

turnBy(110)

forward(.5 , 3)

backward(.5, 3)

turnBy(-40)

forward(.5 ,2.5)

turnBy(40)

backward(.5, 2.5)

turnBy(-55)

forward(.5, 3)

backward(.5, 3)

turnBy(-55)

forward(.5,1.3)

def drawX():

turnBy(60)

forward(.5,3.4)

backward(.5,1.7)

turnBy(60)

forward(.5,1.7)

backward(.5,3.4)

turnBy(-120)

forward(.5,1)

def drawY():

forward(.5,1.5)

turnBy(90)

forward(.5,2.2)

turnBy(30)

forward(.5,2)

backward(.5,2)

turnBy(-60)

forward(.5,2)

backward(.5,2)

turnBy(30)

backward(.5, 2.2)

turnBy(-90)

forward(.5,1.5)

def drawZ():

from Myro import \*

forward(.25,1)

turnBy(60)

forward(.5,4)

turnBy(-60)

backward(.5,2.4)

forward(.5,2.4)

turnBy(60)

backward(.5,4)

turnBy(-60)

forward(.5,4)

def draw\_():

forward(0.5,2)

def write(letter):

if letter == 'A':

drawA()

elif letter == 'B':

drawB()

elif letter == 'C':

drawC()

elif letter == 'D':

drawD()

elif letter == 'E':

drawE()

elif letter == 'F':

drawF()

elif letter == 'G':

drawG()

elif letter == 'H':

drawH()

elif letter == 'I':

drawI()

elif letter == 'J':

drawJ()

elif letter == 'K':

drawK()

elif letter == 'L':

drawL()

elif letter == 'M':

drawM()

elif letter == 'N':

drawN()

elif letter == 'O':

drawO()

elif letter == 'P':

drawP()

elif letter == 'Q':

drawQ()

elif letter == 'R':

drawR()

elif letter == 'S':

drawS()

elif letter == 'T':

drawT()

elif letter == 'U':

drawU()

elif letter == 'V':

drawV()

elif letter == 'W':

drawW()

elif letter == 'X':

drawX()

elif letter == 'Y':

drawY()

elif letter == 'Z':

drawZ()

elif letter == ' ':

draw\_()

def draw(word):

speak(str(word))

word = str.upper(word)

for x in str(word):

forward(.2,1)

speak(x)

write(x)

beep(1/8, 587.3)

beep(1/8, 659.3)

beep(1/8, 698.5)

beep(1/8, 784)

beep(1/4, 659.3)

beep(1/8, 523.3)

beep(3/8, 587.3)

word = raw\_input("What do you want me to draw? ")

draw(word)