import graphics as g,time,math,random

clist=['green','blue','yellow']

disp=[None for x in range(11)]

evals=[None for x in range(5)]

class Player:

"""this is player"""

def \_\_init\_\_(self):

self.x=None

self.y=None

self.pos=None

self.img=None

self.score=0

self.name=None

class building:

def \_\_init\_\_(self):

self.height=None

self.pos=None

def draw(self, pos, height, win):

""" position, height, gaphWin """

self.height=height

self.pos=pos

img=g.Rectangle(g.Point((self.pos-1)\*64.0,0),

g.Point(self.pos\*64.0,self.height))

img.setFill(clist[random.randint(0,2)])

img.draw(win)

class Ball:

def \_\_init\_\_(self):

self.x=None

self.y=None

self.img=None

class enter:

def \_\_init\_\_(self, x, y, width, win):

""" x, y, width, gaphWin """

self.x=x

self.y=y

self.img = g.Entry(g.Point(self.x,self.y),width)

self.img.draw(win)

def gettext(self, si, defa, win):

""" 'string' or 'number', default value """

if si=='s':

self.text = returntext(win)

if self.text=='':

self.text = defa

if si=='n':

try:

check = eval(returntext(win))

self.text = check

except SyntaxError:

self.text = defa

except NameError:

self.text = defa

class display:

def \_\_init\_\_(self, x, y, size, color, text, win):

self.x=x

self.y=y

self.img=g.Text(g.Point(self.x,self.y),text)

self.img.setSize(size)

self.img.setFill(color)

self.img.draw(win)

def put(obj, x, y, win):

obj.img.undraw()

obj.x=x

obj.y=y

obj.img = g.Circle(g.Point(obj.x,obj.y),5)

obj.img.setFill('red')

obj.img.draw(win)

def explosion(x, y, win):

crl=['red', 'orange', 'yellow']

for rad in range(1, 45):

for counr in range(int(math.pi\*rad\*rad/16)):

ranx = random.randint(-rad,+rad)

rany = math.sqrt(rad\*\*2-ranx\*\*2)\*((-1)\*\*(random.randint(0,1)))

p = g.Point(x+ranx,y+rany)

p.setFill(crl[random.randint(0,2)])

p.draw(win)

def returntext(win):

a=str()

b = win.getKey()

while b != 'Return':

if b != 'BackSpace':

a = a + b

else:

a = a[0:(len(a)-1)]

b=win.getKey()

return(a)

def celebrate(obj,tie=False):

win=g.GraphWin('Celebration',500,300)

wi=g.Text(g.Point(250,150),obj.name +''+' Wins!!!')

if tie:

wi=g.Text(g.Point(250,150),'It\'s a tie!!!!\n\n\nPress any key to continue')

wi.setFill("red")

wi.setSize(20)

wi.draw(win)

win.getKey()

win.close()

def check(p, i1, i2):

if p == 0:

return i1

else:

return i2

def askagain():

win=g.GraphWin('Celebration',500,300)

win.setBackground('black')

a=g.Text(g.Point(250,150),"play again(y/n)")

a.setSize(15)

a.setFill('red')

a.draw(win)

b = enter(350, 150, 2, win)

b.gettext('s', 'n', win)

if b.text == 'y':

win.close()

main()

else:

a.undraw()

b.move(1000,1000)

for y in range(0,300):

c=g.Text(g.Point(250,y),'Made By: Arihant Vashista')

c.setSize(15)

c.setFill('red')

c.draw(win)

time.sleep(0.01)

c.undraw()

win.close()

def main():

rcount = 1

pchance = 0

win=g.GraphWin('initial',500,200)

win.setCoords(0,0,5,5)

disp[0] = display(2, 4, 15, 'black', "Player 1(default:Player 1)", win)

disp[1] = display(2, 3, 15, 'black', "Player 2(default:Player 2)", win)

disp[2] = display(2, 2, 15, 'black', "No of Rounds(default:5)", win)

evals[0] = enter(4, 4, 10, win)

evals[0].gettext('s','Player 1',win)

evals[1] = enter(4, 3, 10, win)

evals[1].gettext('s','Player 2',win)

evals[2] = enter(4, 2, 2, win)

evals[2].gettext('n',5,win)

win.close()

no\_of\_rounds = evals[2].text

build=[building() for x in range(10)]

player=[Player() for x in range(2)]

ball=Ball()

player[0].name = evals[0].text

player[1].name = evals[1].text

while rcount<=no\_of\_rounds:

c1=True

win = g.GraphWin('GorPyt',640,480)

win.setCoords(0,0,640,480)

win.setBackground('black')

disp[3] = display(83.2, 456.0, 15, 'yellow', player[0].name, win)

disp[4] = display(546.8, 456.0, 15, 'yellow', player[1].name, win)

disp[5] = display(320.0, 432.0, 20, 'red', 'Round:'+str(rcount), win)

player[0].pos = random.randint(2,3)

player[1].pos = random.randint(8,9)

for x in range(10):

build[x].pos = x+1

build[x].height = 48.0\*random.randint(2,6)

build[x].draw(x+1, build[x].height, win)

disp[6] = display(320.0, 48.0, 20, 'red', '<--- Score --->', win)

disp[7] = display(83.2, 48.0, 20, 'red', str(player[0].score), win)

disp[8] = display(546.8, 48.0, 20, 'red', str(player[1].score), win)

for x in range(2):

for y in range(10):

if player[x].pos == build[y].pos:

player[x].x = y\*64+check(x,20,44)

player[x].y = build[y].height+22.5

player[x].img = g.Image(g.Point(player[x].x,player[x].y),

'data\\p'+str(x+1)+'.gif')

player[x].img.draw(win)

ball.x = player[pchance].x

ball.y = player[pchance].y

ball.img = g.Circle(g.Point(ball.x,ball.y),5)

ball.img.setFill('red')

ball.img.draw(win)

while c1:

c2=True

t=0

disp[9]=display(check(pchance, 83.2,545), 427, 15,'red', 'Angle', win)

disp[10]=display(check(pchance, 83.2,545), 400, 15,'red', 'Velocity', win)

evals[3]=enter(check(pchance, 125, 586.8), 427, 3, win)

evals[3].gettext('n',0,win)

evals[4]=enter(check(pchance, 145, 606.8), 400, 3, win)

evals[4].gettext('n',0, win)

angle = evals[3].text

velocity = evals[4].text

angle = math.radians(angle)

vx = velocity\*math.cos(angle)\*check(pchance, 1, -1)

vy = velocity\*math.sin(angle)

evals[3].img.undraw()

evals[4].img.undraw()

disp[9].img.undraw()

disp[10].img.undraw()

while c2:

dx = player[pchance].x+(vx\*t)

dy=player[pchance].y+(vy\*t-(9.8\*t\*t/2))

put(ball, dx , dy, win)

if ball.x+5 >= 640 or ball.x-5 <=0:

c2 = False

for cx in range(-5, 5):

cy=math.sqrt(25 - cx\*\*2)

for bc in range(10):

if bc == int((ball.x + 5)/64) and ball.y-cy<build[bc].height:

time.sleep(0.05)

c2 = False

if ball.x\*check(pchance,1,-1) + 5> player[check(pchance,1,0)].x\*check(pchance,1,-1) - 15:

if ball.x\*check(pchance,1,-1) - 5<player[check(pchance,1,0)].x\*check(pchance,1,-1) + 15:

if ball.y - cy < player[check(pchance,1,0)].y + 22.5:

c2= False

c1 = False

explosion(player[check(pchance,1,0)].x,

player[check(pchance,1,0)].y, win)

player[pchance].score = player[pchance].score+1

break

break

break

time.sleep(0.01)

t=t+0.1

pchance=check(pchance, 1, 0)

put(ball,player[pchance].x,player[pchance].y,win)

time.sleep(1)

win.close()

rcount = rcount+1

if player[0].score>player[1].score:

celebrate(player[0])

elif player[1].score>player[0].score:

celebrate(player[1])

else:

celebrate(player[0],tie=True)

main()

askagain()