screen = t.Screen()

## set up background color

screen.bgcolor('lightgreen')

## set screen title

screen.title("Rick's Program")

#set up player

player = t.Turtle()

player.color('blue')

player.shape('triangle')

player.penup()

## set up Goal

goal = t.Turtle()

goal.color('red')

goal.shape('circle')

goal.penup()

goal.setpos(-100,100)

speed = 1

## Define Function

def turnleft():

player.lt(30)

def turnright():

player.rt(30)

def speed5():

global speed

speed = 5

def speed1():

global speed

speed = 1

##keyboard Binding

t.listen()

t.onkey(turnleft, 'Left')

t.onkey(turnright, 'Right')

t.onkeypress(speed5, 'Up')

t.onkeyrelease(speed1, 'Up')

while True:

player.forward(speed)

d = np.sqrt((player.xcor()-goal.xcor())\*\*2

+(player.ycor()-goal.ycor())\*\*2)

if d<20:

goal.setpos(random.randint(-300,300)

,random.randint(-300,300))

**Continue:**

**1.Drawing a border & do boundary checking**

**2.Multiple Goals**

**3. change collision into function and call for every goal**

**4. add score**

**5. register shape**

**6. change background shape**

**7. add sound**

**8. count lifes (if have time)**

screen = t.Screen()

## set up background color

screen.bgcolor('lightgreen')

## set screen title

screen.title("Rick's Program")

**## Change background pic**

**screen. bgpic("kbgame-bg.gif")**

**## more smooth**

**screen.tracer(3)**

**## Register Shape**

**screen.register\_shape("mario.gif")**

##Draw a Border

**border = t.Turtle()**

**border.color(“white”)**

**border.penup()**

**border.setposition(-300,-300)**

**border.pendown()**

**border.pensize(3)**

**for side in range(4):**

**border.fd(600)**

**border.lf(90)**

**border.hideturtle()**

**# Set pen for score and life**

**pen1 = turtle.Turtle()**

**pen1.color("white")**

**pen1.hideturtle()**

**pen2 = turtle.Turtle()**

**pen2.color("white")**

**pen2.hideturtle()**

#set up player

player = t.Turtle()

player.color('blue')

player.shape**("mario.gif")**

player.penup()

player.speed(0)

**## create Goals**

**num\_goals = 10**

**goals = []**

**for count in range(num\_goals):**

**goals.append(t.Turtle())**

**goals[count].color("red")**

**goals[count].shape("circle")**

**goals[count].penup()**

**goals[count].speed(0)**

**goals[count].setposition(random.randint(-300, 300), random.randint(-300, 300))**

speed = 1

## Define Function

def turnleft():

player.lt(30)

def turnright():

player.rt(30)

def speed5():

global speed

speed = 5

def speed1():

global speed

speed = 1

**def isCollision(t1, t2):**

**d = math.sqrt(math.pow(t1.xcor()-t2.xcor(),2) + math.pow(t1.ycor()-t2.ycor(),2))**

**if d < 20:**

**return True**

**else:**

**return False**

##keyboard Binding

t.listen()

t.onkey(turnleft, 'Left')

t.onkey(turnright, 'Right')

t.onkeypress(speed5, 'Up')

t.onkeyrelease(speed1, 'Up')

while True:

player.forward(speed)

**#Boundary Checking**

**if player.xcor() > 300 or player.xcor() < -300:**

**player.right(180)**

**os.system("afplay bounce.mp3&")**

**wall = wall-1**

**pen.undo()**

**pen.penup()**

**pen.hideturtle()**

**pen.setposition(290, 310)**

**wallstring = "live: %s" %wall**

**pen2.write(wallstring, False, align="left", font=("Arial",14, "normal"))**

**if wall<=0:**

**break**

**#Boundary Checking**

**if player.ycor() > 300 or player.ycor() < -300:**

**player.right(180)**

**os.system("afplay bounce.mp3&")**

**wall = wall-1**

**pen.undo()**

**pen.penup()**

**pen.hideturtle()**

**pen.setposition(290, 310)**

**wallstring = "live: %s" %wall**

**pen2.write(wallstring, False, align="left", font=("Arial",16, "normal"))**

**if wall<=0:**

**break**

**#Move the goal**

for count in range(maxGoals):

goals[count].forward(3)

**#Boundary Checking**

if goals[count].xcor() > 290 or goals[count].xcor() < -290:

goals[count].right(180)

os.system("afplay bounce.mp3&")

**#Boundary Checking**

if goals[count].ycor() > 290 or goals[count].ycor() < -290:

goals[count].right(180)

os.system("afplay bounce.mp3&")

**#Collision checking**

if isCollision(player, goals[count]):

goals[count].setposition(random.randint(-300, 300), random.randint( 300, 300))

goals[count].right(random.randint(0,360))

os.system("afplay collision.mp3&")

score += 1

#Draw the score on the screen

pen1.undo()

pen1.penup()

pen1.hideturtle()

pen1.setposition(-290, 310)

scorestring = "Score: %s" %score

pen1.write(scorestring, False, align="left", font=("Arial",16, "normal"))

wn.bgcolor("black")

final = turtle.Turtle()

final.hideturtle()

final.color('white')

fail\_string = 'Sorry, you have died. You got total %s points'%score

final.write(fail\_string, False, align="center", font=("Arial",20, "normal"))

screen.exitonclick()