Computer Graphics Project

Pikachu

Submitted By :

19UCS177 Saurav Somani
19UCS003 Shreyansh Jain
19UCS176 Rishabh Jain

Submitted To : Dr. Aloke Datta

Code : https://github.com/Saurav-Somani/CG_Project

Aim

The objective of this project is to make a cartoon character using the concepts and knowledge of Computer graphics.

Approach

We have made Pikachu using python. We have used the 'Turtle' library of python. Using several inbuilt functions of the library we have made a basic design of the cartoon character Pikachu. We have used Turtle because of its immense functionalities and the methods it provides. Also it handles all the trivial tasks very easily.

Inbuilt Functions description

Let's discuss the general functions used in the code which are inbuilt functions in Turtle library.

- penup() : Pull the pen up – no drawing when moving.
- pendown() : Pull the pen down – drawing when moving.
- goto(x,y) : Move turtle to an absolute position. If the pen is down, draw a line. Do not change the turtle's orientation.
- fillcolor(code): This takes the color code as an argument and fills the same color in the object drawn following it.
- begin_fill() : To be called just before drawing a shape to be filled.
- end_fill() : Fill the shape drawn after the last call to begin_fill().

- seth(to_angle) : Set the orientation of the turtle to to_angle. Some directions in standard mode are 0-east,90-north,180-west,270-south.

- circle(radius,extend,steps) : Draw a circle with given radius. The center is radius units left of the turtle; extent – an angle – determines which part of the circle is drawn. If extent is not given, draw the entire circle. If the extent is not a full circle, one endpoint of the arc is the current pen position. Draw the arc in counterclockwise direction if radius is positive, otherwise in clockwise direction. Finally the direction of the turtle is changed by the amount of extent.

- fd(distance) : Move the turtle forward by the specified distance, in the direction the turtle is headed.

- left(angle) : Turn turtle left by angle(degrees by default) units.

- back(distance) : Move the turtle backward by distance, opposite to the direction the turtle is headed. Do not change the turtle's heading.

- pensize(width) : Set the line thickness to width or return it.

- speed(x) : Set the turtle's speed to an integer value in the range 0..10

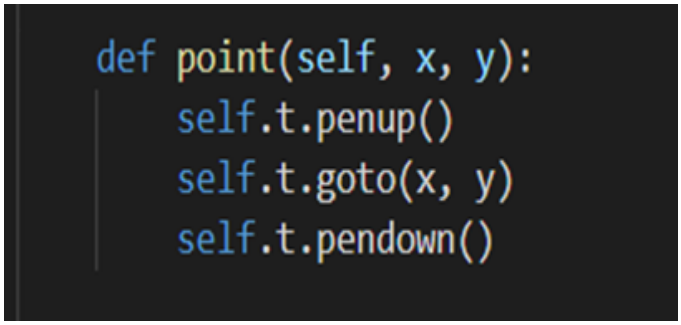
- bgcolor(color) : It sets the background color.

- setup(width,height) : It is used to set the the dimensions of the screen

Drawing Functions

1. **point(self, x, y)**

   In this function we have to move the pen to point(x,y) without drawing anything and put the pen down.
   Inbuilt functions of turtle used penup(),goto(),pendown().



2. **left_eye(self , x, y)**

This is the function for making the left eye. Here all the detailing and outlining of the function is there.It comprises fillcolor(), circle(),begin_fill(),end_fill() and point(x,y).

Code:

```
def left_eye(self, x, y):
    self.point(x, y)
    t = self.t
    t.seth(0)
    t.fillcolor('#333333')
    t.begin_fill()
    t.circle(22)
    t.end_fill()

    self.point(x, y + 10)
    t.fillcolor('#000000')
    t.begin_fill()
    t.circle(10)
    t.end_fill()

    self.point(x + 6, y + 22)
    t.fillcolor('#ffffff')
    t.begin_fill()
    t.circle(10)
    t.end_fill()
```

Output:

3. **right_eye(self , x, y)**

This is the function for making the right eye. Here all the detailing and outlining of the function is there.It comprises fillcolor(), circle(),begin_fill(),end_fill() and point(x,y).

Code:

```
def right_eye(self, x, y):
    self.point(x, y)
    t = self.t
    t.seth(0)
    t.fillcolor('#333333')
    t.begin_fill()
    t.circle(22)
    t.end_fill()

    self.point(x, y + 10)
    t.fillcolor('#000000')
    t.begin_fill()
    t.circle(10)
    t.end_fill()

    self.point(x - 6, y + 22)
    t.fillcolor('#ffffff')
    t.begin_fill()
    t.circle(10)
    t.end_fill()
```

Output:

4. **mouth(self, x, y)**

It is the function used to make the mouth of Pikachu. It uses a loop to make the outer upper boundary of the mouth. It comprises circle(),left(),right(),fill_color(),begin_fill(),end_fill(),fd(),seth().

Code:

```
def mouth(self, x, y):
    self.point(x, y)
    t = self.t

    t.fillcolor('#88141D')
    t.begin_fill()

    t.seth(190)
    a = 0.7
    for i in range(28):
        a += 0.1
        t.right(3)
        t.fd(a)

    self.point(x, y)
    t.seth(10)
    a = 0.7
    for i in range(28):
        a += 0.1
        t.left(3)
        t.fd(a)

    t.seth(10)
    t.circle(50, 15)
    t.left(180)
    t.circle(-50, 15)
    t.circle(-50, 40)
    t.seth(233)
    t.circle(-50, 55)
    t.left(180)
    t.circle(50, 12.1)
    t.end_fill()
```

Output:

5. **nose(self , x, y)**

It is the function responsible for drawing the nose of the character. It comprises of point(),seth(),back(),fd()

Code:

```
def nose(self, x, y):
    self.point(x,y)
    t=self.t
    t.seth(8)
    t.fd(4)
    t.back(8)
```

Output:

6. **draw_the_character(self)**

This is the fundamental function responsible for drawing the outline of the whole character. It is responsible for the color of the whole character. This is the function responsible for calling the above all mentioned functions.

Code :

```
t = self.t
```

```
    t.fillcolor('#F6D02F')
    t.begin_fill()

    #Right cheek
    t.penup()
    t.circle(130, 40)
    t.pendown()
    t.circle(100, 105)
    t.left(180)
    t.circle(-100, 5)

    #Right ear
    t.seth(20)
    t.circle(300, 30)
    t.circle(30, 50)
    t.seth(190)
    t.circle(300, 36)

    #Upper head
    t.seth(150)
    t.circle(150, 70)

    #Left ear
    t.seth(140)
    t.circle(300, 30)
    t.circle(30, 50)
    t.seth(310)
    t.circle(300, 33)

    #Left cheek
    t.seth(240)
    t.circle(105, 95)
    t.left(180)
    t.circle(-105, 5)

    #Left hand
    t.seth(210)
    t.circle(500, 18)
    t.seth(200)
```

```
        t.fd(10)
        t.seth(280)
        t.fd(7)
        t.seth(210)
        t.fd(10)
        t.seth(300)
        t.circle(10, 80)
        t.seth(220)
        t.fd(10)
        t.seth(300)
        t.circle(10, 80)
        t.seth(240)
        t.fd(12)
        t.seth(0)
        t.fd(13)
        t.seth(240)
        t.circle(10, 70)
        t.seth(10)
        t.circle(10, 70)
        t.seth(10)
        t.circle(300, 18)

        #Left part of body
        t.seth(75)
        t.circle(500, 8)
        t.left(180)
        t.circle(-500, 15)
        t.seth(250)
        t.circle(100, 65)

        #Left leg
        t.seth(320)
        t.circle(100, 5)
        t.left(180)
        t.circle(-100, 5)
        t.seth(220)
        t.circle(200, 20)
        t.circle(20, 70)

        t.seth(60)
```

```
        t.circle(-100, 20)
        t.left(180)
        t.circle(100, 20)
        t.seth(300)
        t.circle(10, 70)

        t.seth(60)
        t.circle(-100, 20)
        t.left(180)
        t.circle(100, 20)
        t.seth(10)
        t.circle(100, 60)

        #Bottom part of body
        t.seth(330)
        t.circle(150, 60)

        #Right leg
        t.seth(290)
        t.circle(100, 55)
        t.circle(10, 50)

        t.seth(120)
        t.circle(100, 20)
        t.left(180)
        t.circle(-100, 20)

        t.seth(0)
        t.circle(10, 50)

        t.seth(110)
        t.circle(100, 20)
        t.left(180)
        t.circle(-100, 20)

        t.seth(30)
        t.circle(20, 50)

        t.seth(100)
        t.circle(100, 40)
```

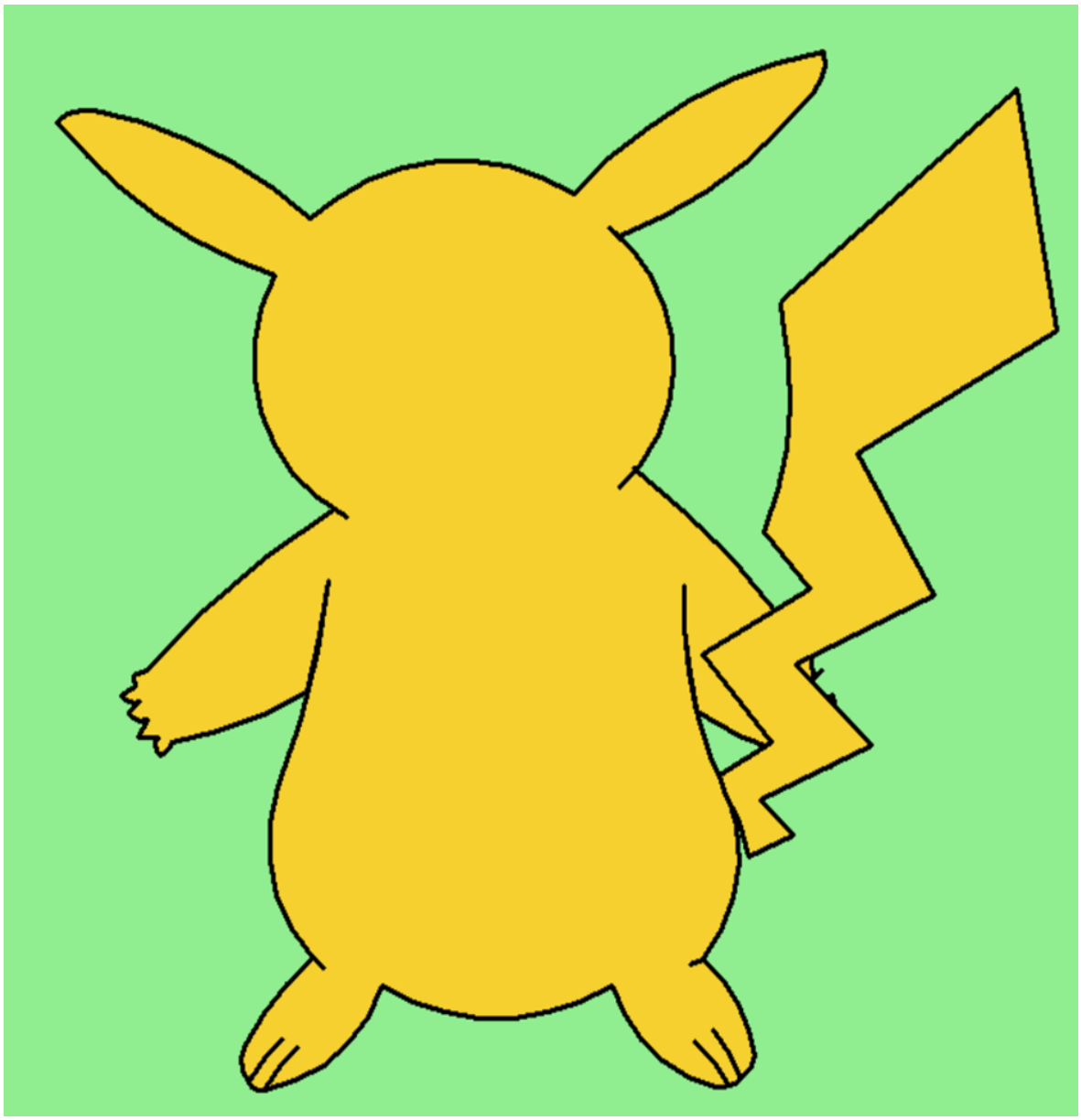
```
    #Right part of body
    t.seth(200)
    t.circle(-100, 5)
    t.left(180)
    t.circle(100, 5)
    t.left(30)
    t.circle(100, 75)
    t.right(15)
    t.circle(-300, 21)
    t.left(180)
    t.circle(300, 15)

    #Right hand
    t.seth(330)
    t.circle(300, 18)
    t.seth(100)
    t.circle(10, 70)
    t.seth(50)
    t.fd(13)
    t.seth(100)
    t.circle(10, 70)
    t.seth(40)
    t.fd(8)
    t.circle(10, 80)
    t.seth(200)
    t.fd(10)
    t.seth(70)
    t.circle(10, 80)
    t.seth(40)
    t.fd(10)
    t.seth(210)
    t.fd(7)
    t.seth(100)
    t.fd(10)
    t.seth(125)
    t.circle(500, 24.25)
    t.seth(210)
    t.end_fill()
```

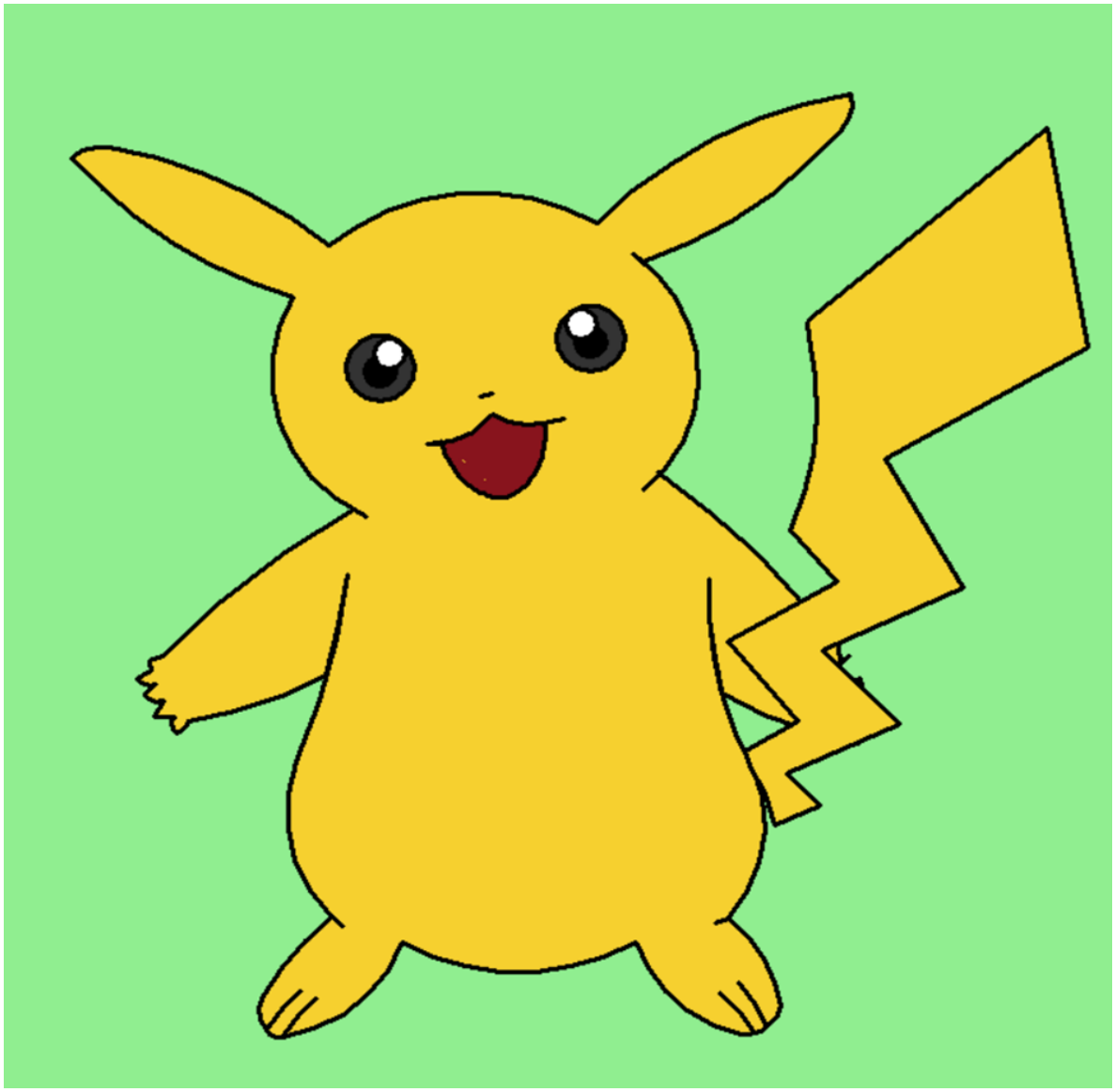
```
        #Tail
        self.point(190, 145)
        t.fillcolor('#F6D02F')
        t.begin_fill()
        t.seth(40)
        t.fd(200)
        t.seth(-80)
        t.fd(150)
        t.seth(210)
        t.fd(150)
        t.left(90)
        t.fd(100)
        t.right(95)
        t.fd(100)
        t.left(110)
        t.fd(70)
        t.right(110)
        t.fd(80)
        t.left(110)
        t.fd(30)
        t.right(110)
        t.fd(32)

        t.right(106)
        t.circle(100, 25)
        t.right(15)
        t.circle(-300, 2)
        t.seth(30)
        t.fd(40)
        t.left(100)
        t.fd(70)
        t.right(100)
        t.fd(80)
        t.left(100)
        t.fd(46)
        t.seth(66)
        t.circle(200, 38)
        t.right(10)
        t.fd(10)
        t.end_fill()
```

Output:



Result



Conclusion

Through this project, we learned about various computer graphics concepts and how to implement them. We learned about circle & line drawing algorithms as well as about drawing in Python using the Turtle module.