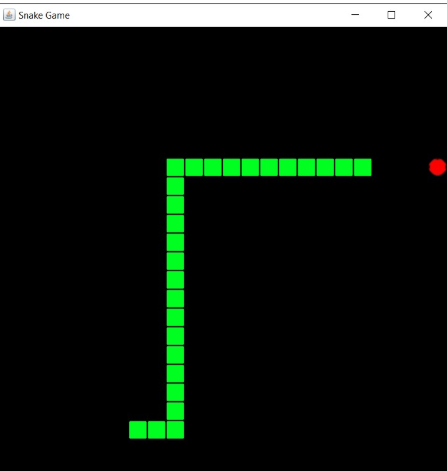
**AIM: - To** Design and Execute a Snake Game for the Toy Problem

**Problem Description: -** To Design a Basic UI and playable Snake game using Py-Libraries and allot various attributes to the Game. The Snake Grows as it catches food, which in turn is decided by a random function. And Fails once it touches itself or the pre-defined boundaries.



**Algorithm: -** 1. Start

2. Initiate the snake

3. Food is randomly placed onto the board

4. The Person needs to move the snake and control its direction

5. Once it meets with the food, a square is added onto the snake

6. Once the snake touches itself or the defined boundaries, the game is over.

7. Goes for unlimited loops

8. The player either looses once the touch is done.

**Program: -**

from turtle import \*

from random import randrange

from freegames import square, vector

food = vector(0, 0)

snake = [vector(10, 0)]

aim = vector(0, -10)

*def* change(x, y):

    "Change snake direction."

    aim.x = x

    aim.y = y

*def* inside(head):

    "Return True if head inside boundaries."

    return -300 < head.x < 290 and -300 < head.y < 290

*def* move():

    "Move snake forward one segment."

    head = snake[-1].copy()

    head.move(aim)

    if not inside(head) or head in snake:

        square(head.x, head.y, 9, 'red')

        update()

        return

    snake.append(head)

    if head == food:

        print('Snake:', len(snake))

        food.x = randrange(-15, 15) \* 10

        food.y = randrange(-15, 15) \* 10

    else:

        snake.pop(0)

    clear()

    for body in snake:

        square(body.x, body.y, 9, 'blue')

    square(food.x, food.y, 9, 'red')

    update()

    ontimer(move, 100)

hideturtle()

tracer(False)

listen()

onkey(*lambda*: change(10, 0), 'Right')

onkey(*lambda*: change(-10, 0), 'Left')

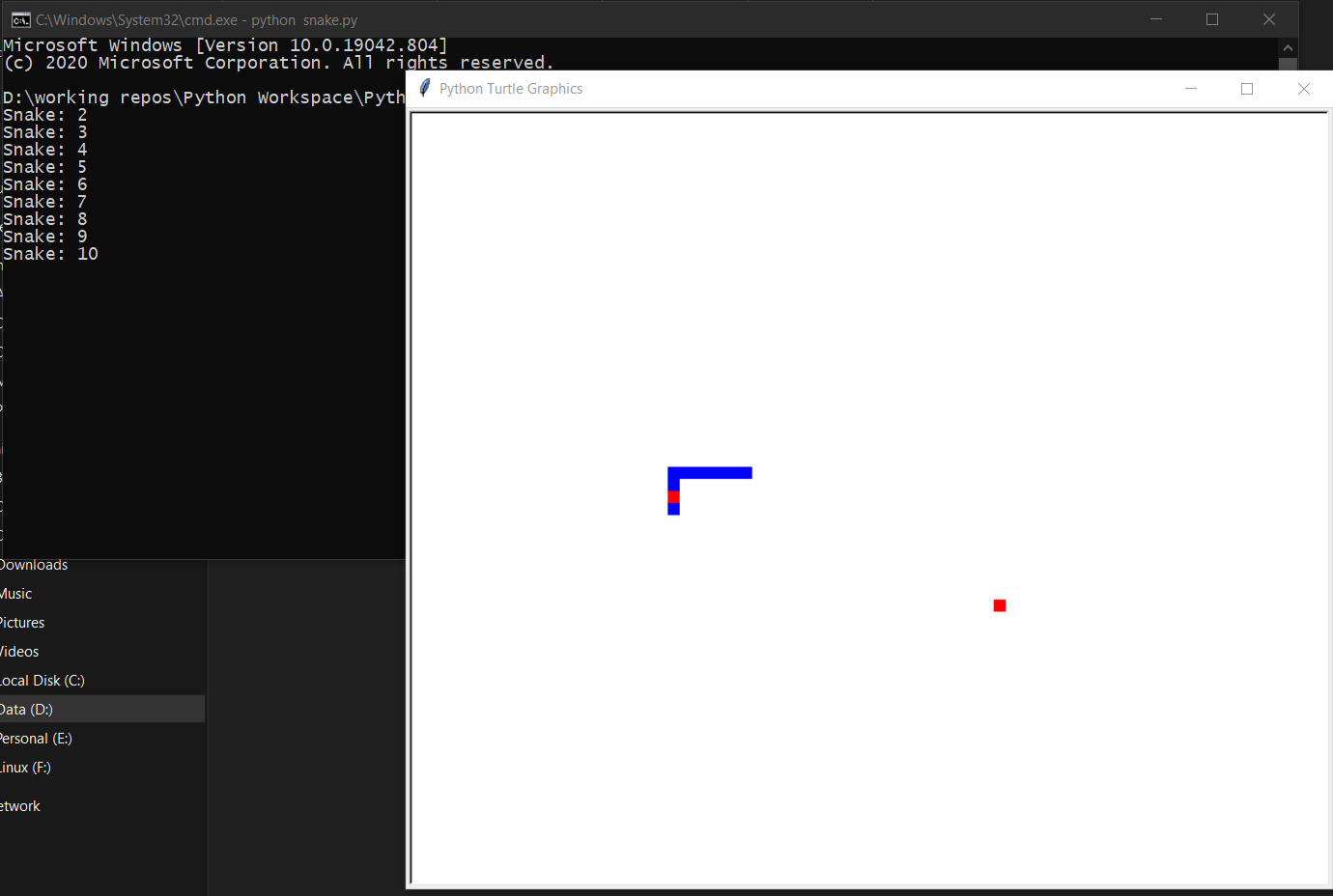
onkey(*lambda*: change(0, 10), 'Up')

onkey(*lambda*: change(0, -10), 'Down')

move()

done(

**Output: -**



**Result: -** Hence a Program to play snake using python with a basic UI was implemented.