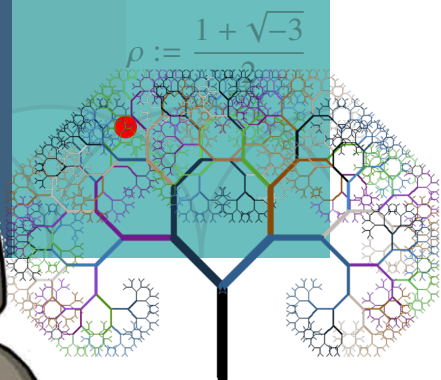


Python-10

Hello Python



目 录

第一章 Python 简介	1
1.1 跨平台的程序语言	1
1.2 系统的安装与执行	1
1.2.1 加法运算	2
1.2.2 减法运算	2
1.2.3 乘法运算	2
1.2.4 幂运算	2
1.2.5 除法和取余数运算	3
1.2.6 处理字符串	3
1.3 Python 文件的编写和执行	3
1.3.1 Python 小测试	4
第二章 Python 绘图	5
2.1 Turtle 绘图	5
2.1.1 绘制线段	5
2.1.2 绘制三角形	6
2.1.3 绘制正方形	6
2.1.4 绘制五边形	7
2.2 习题	9
2.2.1 程序设计题	9
第三章 Python-循环语句绘图	13
3.1 绘制五角星-顺序执行	13
3.2 绘制五角星-for 循环	14
3.3 绘制五角星-while 循环	15

3.4	for 循环绘制圆线条	16
3.5	while 循环绘制圆线条	16
3.6	使用正方形和 for 语句绘制圆形	17
3.7	习题	17
3.7.1	程序设计题	17
3.7.2	程序设计题答案	18
第四章	Python-列表和函数绘图	19
4.1	列表是什么?	19
4.2	通过列表绘制彩色螺线	19
4.3	通过列表绘制五角星	20
4.4	绘制五角星	20
4.5	定义函数	22
4.6	习题	24
4.6.1	程序设计题	24
4.6.2	程序设计题答案	24
第五章	Python-分支语句	29
5.1	分支条件	29
5.2	分支和循环	31
5.3	习题	33
5.3.1	程序设计题	33
5.3.2	程序设计题答案	33
第六章	认识递归	35
6.1	什么是递归	35
6.2	通过递归算法画树	38
6.3	通过递归算法画 coch 雪花	39
第七章	Python-字符串	41
7.1	什么是字符串	41
7.1.1	获取字符串中'o' 字符的个数	41
7.1.2	判断某一个字符串是回文	42
7.1.3	判断某一个字符串的逆序数	43
7.2	字符串转换	44

7.3	字符排序	45
第八章	Python 游戏开发	47
8.1	Pygame 安装	47
8.1.1	猫捉老师	47
8.1.2	画图板	49
8.1.3	游戏的逻辑	49
第九章	贪吃蛇	53
9.1	碰撞检测	53
9.2	蛇的运动	53
9.3	地图	54
9.4	运动速度控制	54
9.5	游戏精灵	56
第十章	超级玛丽	59
10.1	游戏的地图	59
10.2	玛丽的控制	59
10.3	发射子弹	60
10.4	玛丽跳跃	60
10.5	怪物喷火球	62

1

Python 简介

1.1 跨平台的程序语言

Python 是一种跨平台的程序语言，主要操作系统 (Windows、Mac OS、UNIX/Linux) 皆可以安装和使用。跨平台的程序语言意味着用户可以在某一个平台上使用 Python 设计的一个程序，未来这个程序也可以在其他平台上顺利使用。



图 1-1 Windows、Mac OS、UNIX/Linux

1.2 系统的安装与执行

下面将以 Python 3.7x 版本为例做说明，在 Windows 桌面上的 idle(Integrated Development and Learning Environment) 图标，将看到 Python 3.7.0 Shell 窗口。

上述 >>> 符号是提示信息，可以在此输入 Python 指令，下图是一个简单的 print() 函数，目的是输出字符串，单引号或双引号间的文字成为字符串。

由上图可以确定我们成功地执行第一个 Python 的程序实例。

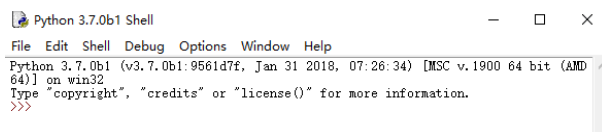


图 1-2 Python 的 IDLE

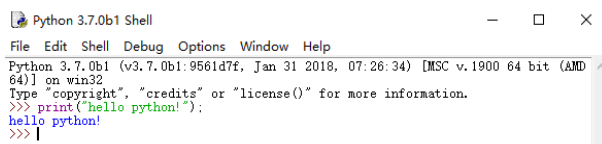


图 1-3 Python 的 Print 函数

1.2.1 加法运算

```
1 a = 2
2 b = 8
3 print(a+b)
```

1.2.2 减法运算

```
1 a = 2
2 b = 8
3 print(b-a)
```

1.2.3 乘法运算

```
1 a = 2
2 b = 8
3 print(a*b)
```

1.2.4 幂运算

```
1 a = 2
2 b = 3
3 print(a*a*a)
```

```
4 print(a**b)
```

1.2.5 除法和取余数运算

```
1 a = 3
2 b = 17
3 print(b/a)
4 print(b%a)
```

1.2.6 处理字符串

```
1 print("Hello Python")
2 print(3*"Hello Python")
```

1.3 Python 文件的编写和执行

如果设计一个程序每次都要在 Python Shell 窗口环境重新输入指令的话，是一件很麻烦的事，当程序设计时，将所设计的程序保存在文档内是一件重要的事。在 Python Shell 窗口可以执行 File/New File，创建一个空白的 Python 文档。

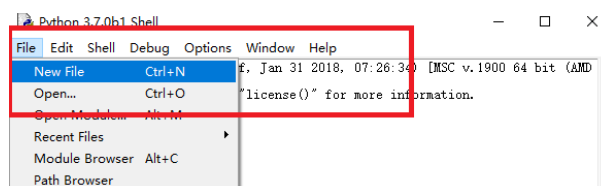


图 1-4 新建 Python 文件

然后创建一个 Untitled 窗口，窗口内容是空白，在空白文档内输入 Python 命令和程序，按 F5 执行保存的 Python 文件。

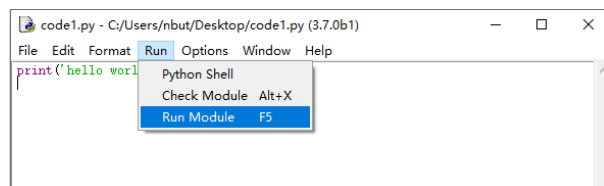


图 1-5 Run Python 文件

1.3.1 Python 小测试

1. 面包店的师傅做个 40 个面包，上午卖掉 12 个面包，下午卖掉 17 个面包，用 Python 程序计算晚上还剩下几个面包？
2. 根据摩尔定律，集成电路芯片上所集成的电路的数目，每隔 18 个月就翻一番，1990 年 1 月份，Intel 生成的芯片有 1000 个晶体管，用 Python 程序计算到 1993 年 1 月份，Intel 生成的芯片包含多少个晶体管？

2

Python 绘图

2.1 Turtle 绘图

表 2-1 Turtle 的主要函数

forward(distance)	fd(distance), 沿着当前方向前进指定距离
backward(distance)	bk(distance), 沿着当前相反方向后退指定距离
right(angle)	改变画笔行进方向为当前方向向右旋转 angle 角度
left(angle)	向左旋转 angle 角度
goto(x,y)	移动到画布中的特定位置 (x, y) 处
setx(x)	修改画笔的横坐标到 x, 纵坐标不变
sety(y)	修改画笔的纵坐标到 y, 横坐标不变
setheading(angle)	设置当前朝向为 angle 角度
home()	设置当前画笔位置为原点, 朝向东
circle(r,e)	绘制一个指定半径 r 和角度 e 的圆或弧形
dot(r,color)	绘制一个指定半径 r 和颜色 color 的圆点
undo()	撤销画笔最后一步动作
speed()	设置画笔的绘制速度, 参数在 0 至 10 之间

2.1.1 绘制线段

- 1、通过 Python 的 Turtle 绘制红色直线。

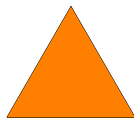
```
1 import turtle
2 import time
3 if __name__=="__main__":
4     t = turtle.Pen();
5     t.shape('turtle');
6     t.pencolor('red');
7     t.penup();
8     t.setpos(0,5);
9     t.pendown();
10    t.fd(100);
11    time.sleep(3);
```



2.1.2 绘制三角形

2、通过 Python 的 Turtle 绘制等边三角形，三角形的内角和为 180° 。

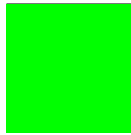
```
1 import turtle
2 import time
3 if __name__=="__main__":
4     t = turtle.Pen()
5     t.shape('turtle')
6     t.pencolor('red')
7     t.fillcolor('orange')
8     t.begin_fill()
9     t.fd(100)
10    t.left(120)
11    t.fd(100)
12    t.left(120)
13    t.fd(100)
14    t.end_fill()
15    time.sleep(3)
```



2.1.3 绘制正方形

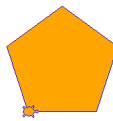
3、通过 Python 的 Turtle 绘制正方形，正方形的内角和为 360° 。

```
1 import turtle
2 import time
3 if __name__=="__main__":
4     t = turtle.Pen()
5     t.shape('turtle')
6     t.fillcolor('green')
7     t.pencolor('blue')
8     t.begin_fill()
9     t.fd(100)
10    t.left(90)
11    t.fd(100)
12    t.left(90)
13    t.fd(100)
14    t.left(90)
15    t.fd(100)
16    t.left(90)
17    t.end_fill()
18    time.sleep(3)
```



2.1.4 绘制五边形

4、绘制五边形，并用橙色填充五边形。



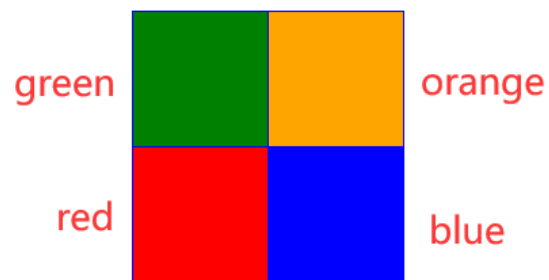
```
1 import turtle
2 import time
3 if __name__=="__main__":
4     t = turtle.Pen();
```

```
5 t.shape('turtle');
6 t.fillcolor('orange');
7 t.pencolor('blue');
8 t.begin_fill();
9 t.fd(100);
10 t.left(72);
11 t.fd(100);
12 t.left(72);
13 t.fd(100);
14 t.left(72);
15 t.fd(100);
16 t.left(72);
17 t.fd(100);
18 t.left(72);
19 t.end_fill();
20 time.sleep(3)
```

2.2 习题

2.2.1 程序设计题

1. 绘制多彩正方形



```
1 import turtle
2 import time
3 if __name__=="__main__":
4     t = turtle.Pen();
5     t.shape('turtle');
6     t.fillcolor('orange');
7     t.pencolor('blue');
8     t.begin_fill();
9     t.fd(100);
10    t.left(90);
11    t.fd(100);
12    t.left(90);
13    t.fd(100);
14    t.left(90);
15    t.fd(100);
16    t.end_fill();
17    t.fillcolor('blue');
18    t.begin_fill()
```



```
19 t.fd(100);
20 t.left(90);
21 t.fd(100);
22 t.left(90);
23 t.fd(100);
24 t.left(90);
25 t.fd(100);
26 t.end_fill()
27 t.fillcolor('red');
28 t.begin_fill()
29 t.fd(100);
30 t.left(90);
31 t.fd(100);
32 t.left(90);
33 t.fd(100);
34 t.left(90);
35 t.fd(100);
36 t.end_fill()
37 t.fillcolor('green');
38 t.begin_fill()
39 t.fd(100);
40 t.left(90);
41 t.fd(100);
42 t.left(90);
43 t.fd(100);
44 t.left(90);
45 t.fd(100);
46 t.end_fill()
47 t.hideturtle()
48 time.sleep(3)
```

2. 使用 pendown()、penup()、goto()、seth() 函数绘制叙利亚的国旗



```
1 import turtle
2 import time
3 if __name__=="__main__":
4     t = turtle.Pen();
5     t.shape('turtle');
6     t.fillcolor('red');
7     t.pencolor('white');
8     t.penup()
9     t.goto(-200,150)
10    t.pendown();
11    t.begin_fill();
12    t.fd(400);
13    t.left(90);
14    t.fd(100);
15    t.left(90);
16    t.fd(400);
17    t.left(90);
18    t.fd(100);
19    t.end_fill();
20    t.fillcolor('white');
21    t.penup()
22    t.goto(-200,50);
23    t.pendown();
24    t.seth(0)
25    t.begin_fill();
26    t.fd(400);
27    t.left(90);
28    t.fd(100);
29    t.left(90);
30    t.fd(400);
31    t.left(90);
32    t.fd(100);
33    t.end_fill()
34    t.fillcolor('black');
35    t.penup()
36    t.goto(-200,-50);
37    t.pendown();
38    t.seth(0)
39    t.begin_fill()
40    t.fd(400);
41    t.left(90);
42    t.fd(100);
43    t.left(90);
44    t.fd(400);
```

```
45     t.left(90);
46     t.fd(100);
47     t.end_fill()
48     t.penup()
49     t.goto(-100,75);
50     t.pendown();
51     t.seth(36)
52     t.pencolor('green')
53     t.fillcolor('green');
54     t.begin_fill();
55     t.fd(50);
56     t.left(144);
57     t.fd(50);
58     t.left(144);
59     t.fd(50);
60     t.left(144);
61     t.fd(50)
62     t.left(144);
63     t.fd(50);
64     t.left(144);
65     t.fd(50);
66     t.end_fill();
67     t.penup()
68     t.goto(75,75);
69     t.pendown();
70     t.seth(36)
71     t.pencolor('green')
72     t.fillcolor('green');
73     t.begin_fill();
74     t.fd(50);
75     t.left(144);
76     t.fd(50);
77     t.left(144);
78     t.fd(50);
79     t.left(144);
80     t.fd(50)
81     t.left(144);
82     t.fd(50);
83     t.left(144);
84     t.fd(50);
85     t.end_fill();
86     t.hideturtle()
87     time.sleep(5)
```

3

Python-循环语句绘图

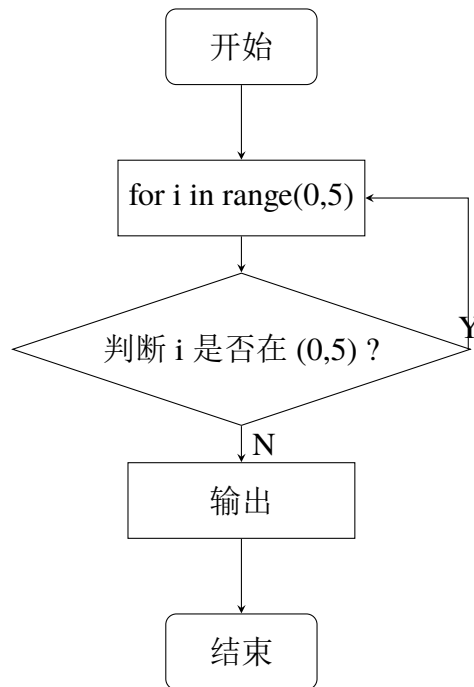
3.1 绘制五角星-顺序执行

1、绘制五角星。

```
1 import turtle
2 import time
3 if __name__=="__main__":
4     t = turtle.Pen()
5     t.shape('turtle')
6     t.fillcolor('red')
7     t.pencolor('red')
8     t.seth(36);
9     t.begin_fill()
10    t.fd(200)
11    t.left(144)
12    t.fd(200)
13    t.left(144)
14    t.fd(200)
15    t.left(144)
16    t.fd(200)
17    t.left(144)
18    t.fd(200)
19    t.left(144)
20    t.end_fill();
21    t.hideturtle();
22    time.sleep(3)
```



3.2 绘制五角星-for 循环



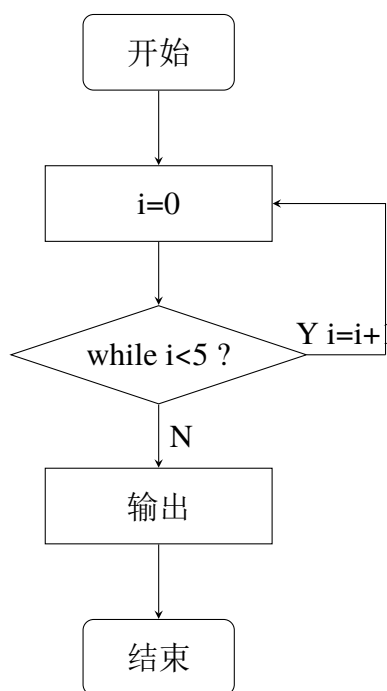
顺序执行绘制五角星的程序，其中第 10 行、第 11 行被重复执行 5 次，因此可以使用循环方式重复执行 5 次。

1、通过 for 循环绘制五角星。

```
1 import turtle
2 import time
3
4 if __name__=="__main__":
5     t = turtle.Pen();
6     t.shape('turtle');
7     t.fillcolor('red');
8     t.pencolor('red');
9     t.seth(36);
10    t.begin_fill();
11    for i in range(5):
12        t.fd(200);
```

```
13     t.left(144);
14     t.end_fill();
15     t.hideturtle();
16     time.sleep(5);
```

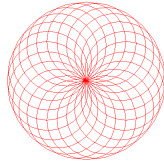
3.3 绘制五角星-while 循环



1、通过 while 绘制五角星。

```
1  import turtle
2  import time
3
4  if __name__=="__main__":
5      t = turtle.Pen();
6      t.shape('turtle');
7      t.fillcolor('red');
8      t.pencolor('red');
9      t.seth(36);
10     i = 0
11     t.begin_fill();
12     while i<5:
13         t.fd(200);
14         t.left(144);
```

```
15     i+=1;
16     t.end_fill();
17     t.hideturtle();
18     time.sleep(3);
```



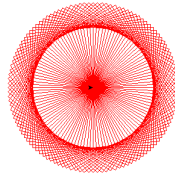
3.4 for 循环绘制圆线条

```
1 import turtle
2 import time
3 if __name__=="__main__":
4     t = turtle.Pen();
5     t.color('red');
6     for i in range(0,360,15):
7         t.seth(i);
8         t.circle(100);
9     time.sleep(3);
```

3.5 while 循环绘制圆线条

```
1 import turtle
2 import time
3 if __name__=="__main__":
4     t = turtle.Pen();
5     t.color('red');
6     i = 0
7     while i<360:
8         t.seth(i);
9         t.circle(100);
10        i+=15;
11    time.sleep(3);
```

3.6 使用正方形和 for 语句绘制圆形

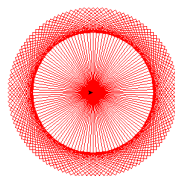


```
1 import turtle
2 import time
3 if __name__=="__main__":
4     t = turtle.Pen();
5     t.pencolor('red');
6     for i in range(0,360,3):
7         t.seth(i);
8         for j in range(4):
9             t.fd(100);
10            t.left(90);
11            time.sleep(3);
```

3.7 习题

3.7.1 程序设计题

1. 仿照 for 循环，使用正方形和 while 语句绘制圆形



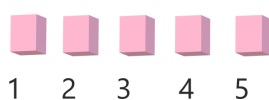
3.7.2 程序设计题答案

```
1 import turtle
2 import time
3 if __name__=="__main__":
4     t = turtle.Pen();
5     t.pencolor('red');
6     i = 0
7     while i<360:
8         t.seth(i);
9         j = 0;
10        while j<4:
11            t.fd(100);
12            t.left(90);
13            j+=1;
14        i+=3;
15    time.sleep(3);
```

4

Python-列表和函数绘图

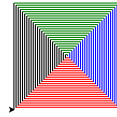
4.1 列表是什么？



4.2 通过列表绘制彩色螺线

```
1 import turtle
2 import time
3 if __name__=="__main__":
4     colors = ['red','blue','green','black'];
5     t = turtle.Pen();
6     t.speed(2);
7     t.pensize(2);
8     for i in range(100):
9         t.pencolor(colors[i%4]);
10        t.fd(2*i);
11        t.left(90);
```

```
12 time.sleep(3);
```



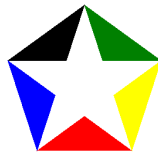
4.3 通过列表绘制五角星

1、绘制边缘是彩色的五角星。

```
1 import turtle
2 import time
3 if __name__=="__main__":
4     t = turtle.Pen();
5     t.shape('turtle');
6     colors = ['red','yellow','green','black','blue'];
7     t.pencolor('red');
8     t.left(36);
9     for i in range(5):
10         t.fillcolor(colors[i]);
11         t.begin_fill();
12         t.pencolor(colors[i]);
13         t.fd(100);
14         t.right(72);
15         t.fd(100);
16         t.left(144);
17         t.end_fill();
18     t.hideturtle();
19     time.sleep(5);
```

4.4 绘制五角星

1、绘制不同颜色的五角星。



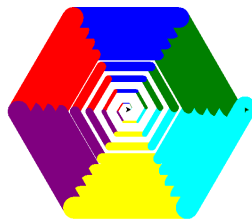
```
1 import turtle
2 import time
3 if __name__=="__main__":
4     t = turtle.Pen();
5     t.shape('turtle');
6     colors = ['cyan','purple','violet','green','red'];
7     t.fillcolor('orange');
8     t.pencolor('red');
9     t.left(36);
10    t.begin_fill();
11    for i in range(0,5):
12        t.pencolor(colors[i]);
13        t.pensize(i+2);
14        t.fd(100);
15        t.right(72);
16        t.fd(100);
17        t.left(144);
18    t.end_fill();
19    t.hideturtle();
20    time.sleep(5);
```



2、绘制不同颜色的六边形。

```
1 import turtle
2 import time
```

```
3 if __name__=="__main__":
4     colors = ['cyan','green','blue','red','purple','yellow'];
5     t = turtle.Pen()
6     t.penup()
7     t.home()
8     t.pendown()
9     turtle.speed(10)
10    t.pencolor(colors[0]);
11    for j in range(50):
12        t.fd(j*5)
13        t.left(60)
14        t.width(j)
15        t.pencolor(colors[j%6])
16    time.sleep(10)
```



4.5 定义函数

1、分别定义五角星、正方形和三角形函数。

```
1 import turtle
2 import time
3
4 def star(t,length,color,fill):
5     t.home();
6     t.left(36);
7     t.pencolor(color);
8     t.fillcolor(fill);
9     t.begin_fill()
10    for i in range(5):
11        t.fd(length);
12        t.right(72);
13        t.fd(length);
14        t.left(144);
```

```
15     t.end_fill();
16
17 def square(t,length,color,fill):
18     t.home();
19     t.fillcolor(fill);
20     t.begin_fill()
21     t.pencolor(color);
22     t.begin_fill()
23     for i in range(4):
24         t.fd(length);
25         t.right(90);
26     t.end_fill()
27
28 def triangle(t,length,color,fill):
29     t.home();
30     t.fillcolor(fill);
31     t.begin_fill()
32     t.pencolor(color);
33     t.begin_fill()
34     for i in range(3):
35         t.fd(length);
36         t.left(120);
37     t.end_fill()
```

2、定义五角星、正方形进而三角形函数之后，通过函数名称调用以上三个函数。

```
38
39 if __name__=="__main__":
40     t = turtle.Pen();
41     t.shape('turtle');
42     colors = ['red','yellow','green','black','blue'];
43     input1 = int(input("input number, 1: star, 2: square, 3: triangle\n"))
44
45     if input1 ==1:
46         star(t,200,colors[2],colors[0]);
47     elif input1 ==2:
48         square(t,200,colors[3],colors[0]);
49     elif input1 ==3:
50         triangle(t,200,colors[4],colors[0]);
51     t.hideturtle();
52     time.sleep(5);
```

4.6 习题

4.6.1 程序设计题

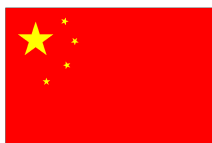
1. 定义长方形函数, 调用函数时, 长为 200, 宽为 100, 填充色为红色。



2. 定义圆和折线, 绘制一个眼镜的图形。



3. 定义五角星函数, 绘制中国国旗。



4.6.2 程序设计题答案

1. 定义长方形函数, 调用函数时, 长为 200, 宽为 100, 填充色为红色。

```
1 import turtle
2 import time
3
```

```
4 def rectangle(t,width,height,color):
5     t.fillcolor(color);
6     t.begin_fill()
7     for i in range(2):
8         t.fd(width)
9         t.left(90)
10        t.fd(height)
11        t.left(90)
12    t.end_fill()
13
14 if __name__=="__main__":
15     t = turtle.Pen();
16     t.pencolor('black');
17     rectangle(t,200,100,'red');
18     t.hideturtle();
19     time.sleep(3);
```

2. 定义圆和折线，绘制一个眼镜的图形。

```
1 import turtle
2 import time
3
4 def glass(pos,t,radius):
5     t.penup()
6     t.goto(pos[0],pos[1])
7     t.pendown()
8     t.circle(radius)
9
10 def jia(pos,t):
11     t.penup()
12     t.goto(pos[0],pos[1])
13     t.pendown()
14     t.seth(60)
15     t.fd(150)
16     t.right(60)
17     t.fd(70)
18
19 def lian(pos,t):
20     t.penup()
21     t.goto(pos[0],pos[1])
22     t.pendown()
23     t.fd(60)
24
```



```
25 if __name__=="__main__":
26     t = turtle.Pen()
27     pos = [(-200,0),(0,0),(-130,70),(-200,140),(67,80)]
28     glass(pos[0],t,70);
29     glass(pos[1],t,70);
30     lian(pos[2],t);
31     jia(pos[3],t);
32     jia(pos[4],t);
33     t.hideturtle();
34     time.sleep(5);
```

3. 定义五角星函数，绘制中国国旗。

```
1 import turtle
2 import time
3
4 def star(t,color,edge):
5     t.color(color)
6     t.fillcolor(color)
7     t.begin_fill()
8     t.right(90)
9     for i in range(5):
10         t.forward(edge)
11         t.right(144)
12     t.end_fill()
13
14
15 if __name__=="__main__":
16     turtle.speed(41)
17     postion = [(-300,240),(-263,163),(-126,211),(-90,140),(-126,
18                                                         64),(-192,30)]
19
20     t = turtle.Pen();
21     t.penup()
22     t.goto(postion[0])
23     t.pendown()
24
25     t.fillcolor('red')
26     t.begin_fill()
27     t.forward(600)
28     t.right(90)
29     t.forward(400)
30     t.right(90)
31     t.forward(600)
```

```
30     t.right(90)
31     t.forward(400)
32     t.end_fill()
33
34     t.penup()
35     t.goto(portion[1])
36     t.pendown()
37     star(t,'yellow',100)
38
39     for i in range(2,6,1):
40         t.penup()
41         t.goto(portion[i])
42         t.pendown()
43         t.color('yellow')
44         star(t,'yellow',20)
45
46 t.hideturtle()
47 time.sleep(10)
```


5

Python-分支语句

5.1 分支条件

1、分段函数

$$y = \begin{cases} 3x + 2 & (x \leq 0) \\ -2x + 6 & (x < 5 \cap x > 0) \\ x^2 + 6 & (x \geq 5) \end{cases} \quad (5.1)$$

```
1 import turtle
2 import time
3
4 def fun1(x):
5     if x <=0:
6         return 3*x+2;
7     elif x<5 and x>0:
8         return -2*x+6;
9     else:
10        return x**2+6
11
12
13 if __name__=="__main__":
14     x = int(input("input number\n"));
15     screen.textinput("NIM", "Name of first player:")
16     t = turtle.Pen()
17     t.pencolor('red')
```

```
18 t.home()
19 t.write('x='+str(x),font=('arial',18,'normal'))
20 t.penup()
21 t.goto(0,-50)
22 t.pendown()
23 t.write('y='+str(fun1(x)),font=('arial',18,'normal'))
24 t.hideturtle()
25 time.sleep(5)
```

x=-3

y=-7

2、输出 100 以内，能被 7 整除的正整数。

```
1 import turtle
2 import time
3
4 def fun1(x):
5     if x%7==0:
6         return True;
7     else:
8         return False;
9
10 if __name__=="__main__":
11     i=1;
12     t = turtle.Pen()
13     t.penup()
14     t.goto(-400,0)
15     t.pendown()
16     while(i<=100):
17         if(fun1(i)==True):
18             t.penup()
19             t.fd(40)
20             t.pendown()
21             t.write(i,font=('arial',18,'normal'))
22             i+=1;
```

```
23 t.hideturtle()
24 time.sleep(5)
```

7 14 21 28 35 42 49 56 63 70 77 84 91 98

5.2 分支和循环

3、输出 0-100 之间所有的质数 (只能被 1 和自身整除的数)。

2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97

```
1 import turtle
2 import time
3
4 def prime(n):
5     i =2;
6     while(i<n):
7         if n%i==0:
8             return False;
9         else:
10            i+=1;
11    return True;
12
13 if __name__=="__main__":
14     i=2;
15     t = turtle.Pen()
16     t.penup()
17     t.goto(-600,0)
18     t.pendown()
19     while(i<=100):
20         if(prime(i)==True):
```

```

21     t.penup()
22     t.fd(40)
23     t.pendown()
24     t.write(i,font=('arial',18,'normal'))
25     i+=1;
26 t.hideturtle()
27 time.sleep(5)

```

4、鸡兔同笼，共有 45 个头，146 只脚。笼中鸡兔各有多少只？

兔子有28



鸡有17



```

1  import turtle
2  import time
3
4  def fun(foot):
5      i = 0;
6      while(i<=36):
7          if((i*4+(45-i)*2)==146):
8              return i;
9          i+=1;
10
11 def fun1(foot):
12     i = 0;
13     while(i<=73):
14         if((i*2+(45-i)*4)==146):
15             return 45-i;
16         i+=1;
17
18 if __name__=="__main__":
19     t = turtle.Pen();
20     t.pencolor('black');
21     t.penup();
22     t.goto(-400,0);

```

```
23 t.pendown();
24 t.write("兔子有"+str(fun1(146)),font=('arial',18,'normal'));
25 t.penup();
26 t.goto(-400,-100);
27 t.pendown();
28 t.write("鸡有"+str(45-fun1(146)),font=('arial',18,'normal'));
29 t.hideturtle();
30 time.sleep(5);
```

5.3 习题

5.3.1 程序设计题

1. 一只螃蟹有 10 只脚；一只蜻蜓有 6 只脚，两对翅膀；一只螳螂有 6 只脚，一对翅膀。现有螃蟹、蜻蜓、螳螂共 37 只，合计有脚 250 只，翅膀 52 对。求螃蟹、蜻蜓、螳螂各有多少只？

螃蟹有7



蜻蜓有22



螳螂有8



5.3.2 程序设计题答案

1. 一只螃蟹有 10 只脚；一只蜻蜓有 6 只脚，两对翅膀；一只螳螂有 6 只脚，一对翅膀。现有螃蟹、蜻蜓、螳螂共 37 只，合计有脚 250 只，翅膀 52 对。求螃蟹、蜻蜓、螳螂各有多少只？

```
1 import turtle
```



```
2 import time
3
4 def fun():
5     i = 0;
6     while(i<=26):
7         j = 52;
8         while(j>=0):
9             if((2*i+j==52) and ((37-i-j)*10+6*i+6*j)==250):
10                 return 37-i-j,i,j;
11             j-=1;
12             i+=1;
13
14
15 if __name__=="__main__":
16     t = turtle.Pen();
17     t.pencolor('black');
18     t.penup();
19     t.goto(-400,0);
20     t.pendown();
21     t.write("螃蟹有"+str(fun()[0]),font=('arial',18,'normal'));
22     t.penup();
23     t.goto(-400,-100);
24     t.pendown();
25     t.write("蜻蜓有"+str(fun()[1]),font=('arial',18,'normal'));
26     t.penup();
27     t.goto(-400,-200);
28     t.pendown();
29     t.write("螳螂有"+str(fun()[2]),font=('arial',18,'normal'));
30     t.hideturtle();
31     time.sleep(5);
```

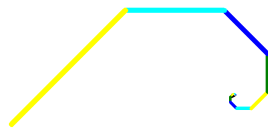
6

认识递归

6.1 什么是递归

1、斐波那契数列

$$\begin{cases} F(1) = 1 \\ F(2) = 1 \\ F(n) = F(n-1) + F(n-2) \quad (n \geq 3, n \in N^*) \end{cases} \quad (6.1)$$



```
1 import turtle
2 import time
3
4 def fabonaci(n):
5     if n == 1:
6         return 1;
7     elif n == 2:
8         return 1;
9     else:
```

```

10     return fabonaci(n-1)+fabonaci(n-2);
11
12 def drawfabonaci(t,n):
13     i = 1
14     colors = ['cyan','yellow','green','blue'];
15     while i<=n:
16         t.fd(fabonaci(i));
17         t.left(45);
18         t.pencolor(colors[i%4]);
19         t.width(i);
20         i+=1
21
22
23 if __name__=="__main__":
24     n = 14;
25     t = turtle.Pen();
26     turtle.speed(0);
27     drawfabonaci(t,n);
28     t.hideturtle();
29     time.sleep(5);

```

2、求和

$$y = \sum_{x=1}^{15} x \quad (6.2)$$



```

1 import turtle
2 import time
3
4 def sigma(n):
5     if n == 1:
6         return 1;
7     else:
8         return sigma(n-1)+n;
9

```

```

10 def drawsigma(t,n):
11     i = 1
12     colors = ['cyan','yellow','green','blue'];
13     while i<=n:
14         t.fd(sigma(i));
15         t.left(45);
16         t.pencolor(colors[i%4]);
17         t.width(i);
18         i+=1
19
20 if __name__=="__main__":
21     n = 15;
22     t = turtle.Pen();
23     turtle.speed(0);
24     drawsigma(t,n);
25     t.hideturtle();
26     time.sleep(5);

```

3、求积

$$y = \prod_{x=1}^{15} x \quad (6.3)$$



```

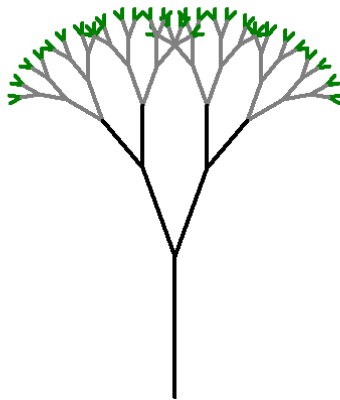
1 import turtle
2 import time
3
4 def product(n):
5     if n == 1:
6         return 1;
7     else:
8         return product(n-1)*n;
9
10 def drawproduct(t,n):
11     i = 1

```

```
12 colors = ['cyan', 'yellow', 'green', 'blue'];
13 while i<=n:
14     t.fd(product(i));
15     t.left(125);
16     t.pencolor(colors[i%4]);
17     t.width(i);
18     i+=1
19
20 if __name__=="__main__":
21     n = 6;
22     t = turtle.Pen();
23     turtle.speed(0);
24     drawproduct(t,n);
25     t.hideturtle();
26     time.sleep(5);
```

6.2 通过递归算法画树

- 1、通过 `draw_tree(t,deep)` 函数画树。

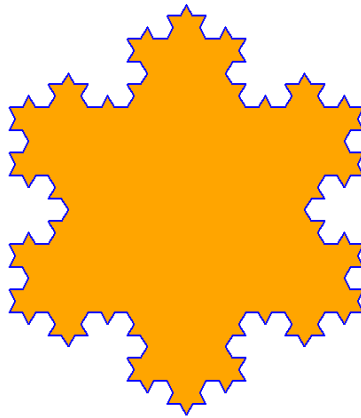


```
1 import turtle
2 import time
3
4 size = 10
5
6 def draw_tree(t,deep):
7     if deep>size:
8         t.fd(deep)
9         t.right(20)
10        draw_tree(t,deep/1.5)
11        t.left(40)
12        draw_tree(t,deep/1.5)
13        t.right(20)
14        if deep<1.5*size:
15            t.pencolor('green')
16        elif deep<5*size:
17            t.pencolor('gray')
18        else:
19            t.pencolor('black')
20        t.bk(deep)
21
22
23 if __name__=="__main__":
24     t = turtle.Pen()
25     t.shape('arrow');
26     t.pensize(4);
27     t.penup()
28     t.goto(0,-200);
29     t.pendown();
30     t.left(90)
31     draw_tree(t,140)
32     t.hideturtle();
33     time.sleep(5);
```

6.3 通过递归算法画 coch 雪花

1、通过 coch(t,size,n) 函数雪花。

```
1 import turtle
2 import time
3
```



```
4 def coch(t,size,n):
5     if n==0:
6         t.fd(size)
7     else:
8         for x in [0,60,-120,60]:
9             t.left(x)
10            coch(t,size/3,n-1)
11
12 if __name__=="__main__":
13     t = turtle.Pen()
14     t.color("blue","orange")
15     t.begin_fill()
16     turtle.setup(600,600)
17     t.penup()
18     t.pensize(2)
19     t.goto(-200,100)
20     t.pendown()
21     a=3
22     coch(t,400,a)
23     t.right(120)
24     coch(t,400,a)
25     t.right(120)
26     coch(t,400,a)
27     t.end_fill()
28     t.hideturtle()
29     time.sleep(5)
```

7

Python-字符串

7.1 什么是字符串

1、字符串就是字符构成的集合，通常包含字母、数字、各种符号，请问各位同学，日常生活中，有哪些字符串？

- ▶ 电话号码
- ▶ 银行卡号
- ▶ 身份证号码

7.1.1 获取字符串中'o' 字符的个数

hello python 2

```
1 import turtle
2 import time
3
4 def number(s,input1):
5     i = 0;
6     sum1 = 0;
7     while(i<len(s)):
8         if s[i]==input1:
```



```

9     sum1+=1;
10    i+=1;
11    return sum1;
12
13 if __name__=="__main__":
14     s = "hello python";
15     input1 = 'o';
16     t = turtle.Pen();
17     t.pencolor('red');
18     t.home();
19     t.write(s,font=('arial',14,'normal'));
20     t.penup()
21     t.fd(150);
22     t.pendown()
23     t.write(number(s,input1),font=('arial',14,'normal'));
24     t.hideturtle();
25     time.sleep(4);

```

7.1.2 判断某一个字符串是回文

2、回文是一个对称的数字、或字符串。

heeh是回文数

test不是回文数

```

1  import turtle
2  import time
3
4  def fun1(s):
5      s1 = ""
6      if len(s)>0:
7          i =0
8          while i<len(s):
9              s1=s[i]+s1
10             i+=1;
11         return s1
12     else:
13         return None
14
15 if __name__=="__main__":
16     s = input("input the number\n");
17     t = turtle.Pen();

```

```

18 turtle.speed(10);
19 t.penup()
20 t.goto(-200,0)
21 t.pendown()
22 t.pencolor('red')
23 if s ==fun1(s):
24     t.write(fun1(s)+"是回文数",font=('arial',18,'normal'));
25 else:
26     t.write(s+"不是回文数",font=('arial',18,'normal'));
27 t.hideturtle()
28 time.sleep(5);

```

7.1.3 判断某一个字符串的逆序数

3、例如 31542 的逆序数为 5。

12345的逆序数为0

```

1 import turtle
2 import time
3
4 def fun1(s):
5     s1 = 0
6     if len(s)>0:
7         i = len(s)-1
8         while i>=1:
9             j=i-1
10            while j>=0:
11                if int(s[i])<int(s[j]):
12                    s1+=1
13                j-=1;
14            i-=1
15        return s1
16    else:
17        return None
18
19 if __name__=="__main__":
20     s = input("input the number\n");
21     t = turtle.Pen();
22     turtle.speed(10);

```

```
23 t.penup()
24 t.goto(-400,0)
25 t.pendown()
26 t.pencolor('red')
27 t.write(s + "的逆序数为" + str(fun1(s)),font=('arial',18,'normal'));
28 t.hideturtle()
29 time.sleep(5);
```

7.2 字符串转换

4、将小写字符串转换为大写字符串。

abcd大写字符串为ABCD

```
1 import turtle
2 import time
3
4 def fun1(s):
5     s1 = ""
6     i = 0
7     while i < len(s):
8         s1 = s1 + chr(ord(s[i]) - 32);
9         i += 1
10    return s1
11
12 if __name__ == "__main__":
13     s = input("input the string\n");
14     t = turtle.Pen();
15     turtle.speed(10);
16     t.penup()
17     t.goto(-400,0)
18     t.pendown()
19     t.pencolor('red')
20     t.write(s + "大写字符串为" + fun1(s),font=('arial',18,'normal'));
21     t.hideturtle()
22     time.sleep(5);
```

7.3 字符排序

5、设计 sort 函数，对字符串进行排序，例如调用 sort('morning') 应返回"gimnnor"。

morning排序后的字符串gimnnor

```
1 import turtle
2 import time
3
4 def sort(x):
5     x = [a for a in x]
6     i = 0
7     while(i<len(x)-1):
8         j = 0
9         while(j<len(x)-i-1):
10             if(x[j]>x[j+1]):
11                 x[j+1],x[j] = x[j],x[j+1]
12             j+=1
13         i+=1
14     return x
15
16 if __name__=="__main__":
17     x = 'morning'
18     t = turtle.Pen();
19     turtle.speed(10);
20     t.penup()
21     t.goto(-400,0)
22     t.pendown()
23     t.pencolor('red')
24     t.write(str(x) + "排序后的字符串"+ ''.join(sort(x)),font=('arial',18,'
normal'));
25     t.hideturtle()
26     time.sleep(5);
```


8

Python 游戏开发

8.1 Pygame 安装

Pygame 不是 python 默认安装的模块，可以从 www.pygame.org 下载 Pygame，选择合适你的操作系统和合适的版本，然后安装就可以了。安装之后，可以用下面的方法确认下有没有安装成功：

```
1 pip install pygame
2 import pygame
3 print(pygame.ver)
4 1.9.1
```

8.1.1 猫捉老师

```
1 import pygame, sys
2 from pygame.locals import *
3 pygame.init()
4 FPS = 30 # frames per second setting
5 fpsClock = pygame.time.Clock()
6 DISPLAYSURF = pygame.display.set_mode((400, 300), 0, 32)
7 pygame.display.set_caption('Animation')
8 WHITE = (255, 255, 255)
9 catImg = pygame.image.load('cat.png')
10 catx = 10
11 caty = 10
```

```
12 direction = 'right'
13 while True: # the main game loop
14     DISPLAYSURF.fill(WHITE)
15     if direction == 'right':
16         catx += 5
17         if catx == 280:
18             direction = 'down'
19     elif direction == 'down':
20         caty += 5
21         if caty == 220:
22             direction = 'left'
23     elif direction == 'left':
24         catx -= 5
25         if catx == 10:
26             direction = 'up'
27     elif direction == 'up':
28         caty -= 5
29         if caty == 10:
30             direction = 'right'
31     DISPLAYSURF.blit(catImg, (catx, caty))
32     for event in pygame.event.get():
33         if event.type == QUIT:
34             pygame.quit()
35             sys.exit()
36     pygame.display.update()
37     fpsClock.tick(FPS)
```

 Animation

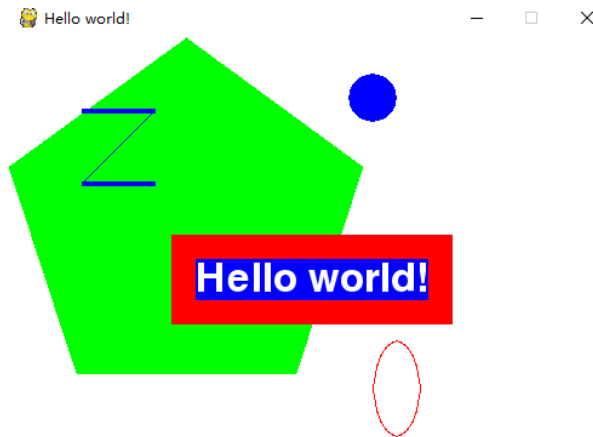


8.1.2 画图板

```
1 import pygame, sys
2 from pygame.locals import *
3 pygame.init()
4 windowSurface = pygame.display.set_mode((500, 400), 0, 32)
5 pygame.display.set_caption('Hello world!')
6 BLACK = (0, 0, 0)
7 WHITE = (255, 255, 255)
8 RED = (255, 0, 0)
9 GREEN = (0, 255, 0)
10 BLUE = (0, 0, 255)
11 basicFont = pygame.font.SysFont(None, 48)
12 text = basicFont.render('Hello world!', True, WHITE, BLUE)
13 textRect = text.get_rect()
14 textRect.centerx = windowSurface.get_rect().centerx
15 textRect.centery = windowSurface.get_rect().centery
16 windowSurface.fill(WHITE)
17 pygame.draw.polygon(windowSurface, GREEN, ((146, 0), (291, 106), (236,
18                                     277), (56, 277), (0, 106)))
19 pygame.draw.line(windowSurface, BLUE, (60, 60), (120, 60), 4)
20 pygame.draw.line(windowSurface, BLUE, (120, 60), (60, 120))
21 pygame.draw.line(windowSurface, BLUE, (60, 120), (120, 120), 4)
22 pygame.draw.circle(windowSurface, BLUE, (300, 50), 20, 0)
23 pygame.draw.ellipse(windowSurface, RED, (300, 250, 40, 80), 1)
24 pygame.draw.rect(windowSurface, RED, (textRect.left - 20, textRect.top
25                                     - 20, textRect.width + 40,
26                                     textRect.height + 40))
27
28 pixArray = pygame.PixelArray(windowSurface)
29 pixArray[480][380] = BLACK
30 del pixArray
31 windowSurface.blit(text, textRect)
32 pygame.display.update()
33 while True:
34     for event in pygame.event.get():
35         if event.type == QUIT:
36             pygame.quit()
37             sys.exit()
```

8.1.3 游戏的逻辑

```
1 import pygame, sys
```

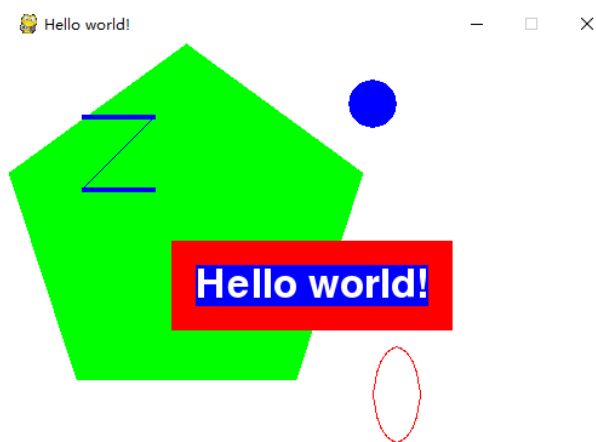



```

2  from pygame.locals import *
3  pygame.init()
4  windowSurface = pygame.display.set_mode((500, 400), 0, 32)
5  pygame.display.set_caption('Hello world!')
6  BLACK = (0, 0, 0)
7  WHITE = (255, 255, 255)
8  RED = (255, 0, 0)
9  GREEN = (0, 255, 0)
10 BLUE = (0, 0, 255)
11 basicFont = pygame.font.SysFont(None, 48)
12 text = basicFont.render('Hello world!', True, WHITE, BLUE)
13 textRect = text.get_rect()
14 textRect.centerx = windowSurface.get_rect().centerx
15 textRect.centery = windowSurface.get_rect().centery
16 windowSurface.fill(WHITE)
17 pygame.draw.polygon(windowSurface, GREEN, ((146, 0), (291, 106), (236,
18                                     277), (56, 277), (0, 106)))
19 pygame.draw.line(windowSurface, BLUE, (60, 60), (120, 60), 4)
20 pygame.draw.line(windowSurface, BLUE, (120, 60), (60, 120))
21 pygame.draw.line(windowSurface, BLUE, (60, 120), (120, 120), 4)
22 pygame.draw.circle(windowSurface, BLUE, (300, 50), 20, 0)
23 pygame.draw.ellipse(windowSurface, RED, (300, 250, 40, 80), 1)
24 pygame.draw.rect(windowSurface, RED, (textRect.left - 20, textRect.top
25                                     - 20, textRect.width + 40,
26                                     textRect.height + 40))
27
28 pixArray = pygame.PixelArray(windowSurface)
29 pixArray[480][380] = BLACK
30 del pixArray

```

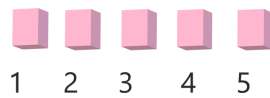
```
27 windowSurface.blit(text, textRect)
28 pygame.display.update()
29 while True:
30     for event in pygame.event.get():
31         if event.type == QUIT:
32             pygame.quit()
33             sys.exit()
```



9

贪吃蛇

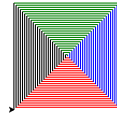
9.1 碰撞检测



9.2 蛇的运动

```
1 import turtle
2 import time
3 if __name__=="__main__":
4     colors = ['red','blue','green','black'];
5     t = turtle.Pen();
6     t.speed(2);
7     t.pensize(2);
8     for i in range(100):
9         t.pencolor(colors[i%4]);
10        t.fd(2*i);
11        t.left(90);
```

```
12 time.sleep(3);
```



9.3 地图

1、绘制边缘是彩色的五角星。

```
1 import turtle
2 import time
3 if __name__=="__main__":
4     t = turtle.Pen();
5     t.shape('turtle');
6     colors = ['red','yellow','green','black','blue'];
7     t.pencolor('red');
8     t.left(36);
9     for i in range(5):
10         t.fillcolor(colors[i]);
11         t.begin_fill();
12         t.pencolor(colors[i]);
13         t.fd(100);
14         t.right(72);
15         t.fd(100);
16         t.left(144);
17         t.end_fill();
18     t.hideturtle();
19     time.sleep(5);
```

9.4 运动速度控制

1、绘制不同颜色的五角星。



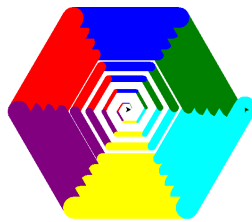
```
1 import turtle
2 import time
3 if __name__=="__main__":
4     t = turtle.Pen();
5     t.shape('turtle');
6     colors = ['cyan','purple','violet','green','red'];
7     t.fillcolor('orange');
8     t.pencolor('red');
9     t.left(36);
10    t.begin_fill();
11    for i in range(0,5):
12        t.pencolor(colors[i]);
13        t.pensize(i+2);
14        t.fd(100);
15        t.right(72);
16        t.fd(100);
17        t.left(144);
18    t.end_fill();
19    t.hideturtle();
20    time.sleep(5);
```



2、绘制不同颜色的六边形。

```
1 import turtle
2 import time
```

```
3 if __name__=="__main__":
4     colors = ['cyan','green','blue','red','purple','yellow'];
5     t = turtle.Pen()
6     t.penup()
7     t.home()
8     t.pendown()
9     turtle.speed(10)
10    t.pencolor(colors[0]);
11    for j in range(50):
12        t.fd(j*5)
13        t.left(60)
14        t.width(j)
15        t.pencolor(colors[j%6])
16    time.sleep(10)
```



9.5 游戏精灵

1、分别定义五角星、正方形和三角形函数。

```
1 import turtle
2 import time
3
4 def star(t,length,color,fill):
5     t.home();
6     t.left(36);
7     t.pencolor(color);
8     t.fillcolor(fill);
9     t.begin_fill()
10    for i in range(5):
11        t.fd(length);
12        t.right(72);
13        t.fd(length);
14        t.left(144);
```

```
15     t.end_fill();
16
17     def square(t,length,color,fill):
18         t.home();
19         t.fillcolor(fill);
20         t.begin_fill()
21         t.pencolor(color);
22         t.begin_fill()
23         for i in range(4):
24             t.fd(length);
25             t.right(90);
26         t.end_fill()
27
28     def triangle(t,length,color,fill):
29         t.home();
30         t.fillcolor(fill);
31         t.begin_fill()
32         t.pencolor(color);
33         t.begin_fill()
34         for i in range(3):
35             t.fd(length);
36             t.left(120);
37         t.end_fill()
```

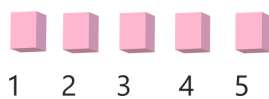
2、定义五角星、正方形进而三角形函数之后，通过函数名称调用以上三个函数。

```
38
39     if __name__=="__main__":
40         t = turtle.Pen();
41         t.shape('turtle');
42         colors = ['red','yellow','green','black','blue'];
43         input1 = int(input("input number, 1: star, 2: square, 3: triangle\n"))
44
45         if input1 ==1:
46             star(t,200,colors[2],colors[0]);
47         elif input1 ==2:
48             square(t,200,colors[3],colors[0]);
49         elif input1 ==3:
50             triangle(t,200,colors[4],colors[0]);
51         t.hideturtle();
52         time.sleep(5);
```


10

超级玛丽

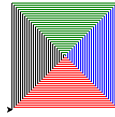
10.1 游戏的地图



10.2 玛丽的控制

```
1 import turtle
2 import time
3 if __name__=="__main__":
4     colors = ['red','blue','green','black'];
5     t = turtle.Pen();
6     t.speed(2);
7     t.pensize(2);
8     for i in range(100):
9         t.pencolor(colors[i%4]);
10        t.fd(2*i);
11        t.left(90);
```

```
12 | time.sleep(3);
```



10.3 发射子弹

- 1、绘制边缘是彩色的五角星。

```
1 import turtle
2 import time
3 if __name__=="__main__":
4     t = turtle.Pen();
5     t.shape('turtle');
6     colors = ['red','yellow','green','black','blue'];
7     t.pencolor('red');
8     t.left(36);
9     for i in range(5):
10         t.fillcolor(colors[i]);
11         t.begin_fill();
12         t.pencolor(colors[i]);
13         t.fd(100);
14         t.right(72);
15         t.fd(100);
16         t.left(144);
17         t.end_fill();
18     t.hideturtle();
19     time.sleep(5);
```

10.4 玛丽跳跃

- 1、绘制不同颜色的五角星。



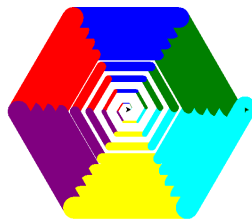
```
1 import turtle
2 import time
3 if __name__=="__main__":
4     t = turtle.Pen();
5     t.shape('turtle');
6     colors = ['cyan','purple','violet','green','red'];
7     t.fillcolor('orange');
8     t.pencolor('red');
9     t.left(36);
10    t.begin_fill();
11    for i in range(0,5):
12        t.pencolor(colors[i]);
13        t.pensize(i+2);
14        t.fd(100);
15        t.right(72);
16        t.fd(100);
17        t.left(144);
18    t.end_fill();
19    t.hideturtle();
20    time.sleep(5);
```



2、绘制不同颜色的六边形。

```
1 import turtle
2 import time
```

```
3 if __name__=="__main__":
4     colors = ['cyan','green','blue','red','purple','yellow'];
5     t = turtle.Pen()
6     t.penup()
7     t.home()
8     t.pendown()
9     turtle.speed(10)
10    t.pencolor(colors[0]);
11    for j in range(50):
12        t.fd(j*5)
13        t.left(60)
14        t.width(j)
15        t.pencolor(colors[j%6])
16    time.sleep(10)
```



10.5 怪物喷火球

1、分别定义五角星、正方形和三角形函数。

```
1 import turtle
2 import time
3
4 def star(t,length,color,fill):
5     t.home();
6     t.left(36);
7     t.pencolor(color);
8     t.fillcolor(fill);
9     t.begin_fill()
10    for i in range(5):
11        t.fd(length);
12        t.right(72);
13        t.fd(length);
14        t.left(144);
```

```
15     t.end_fill();
16
17 def square(t,length,color,fill):
18     t.home();
19     t.fillcolor(fill);
20     t.begin_fill()
21     t.pencolor(color);
22     t.begin_fill()
23     for i in range(4):
24         t.fd(length);
25         t.right(90);
26     t.end_fill()
27
28 def triangle(t,length,color,fill):
29     t.home();
30     t.fillcolor(fill);
31     t.begin_fill()
32     t.pencolor(color);
33     t.begin_fill()
34     for i in range(3):
35         t.fd(length);
36         t.left(120);
37     t.end_fill()
```

2、定义五角星、正方形进而三角形函数之后，通过函数名称调用以上三个函数。

```
38
39 if __name__=="__main__":
40     t = turtle.Pen();
41     t.shape('turtle');
42     colors = ['red','yellow','green','black','blue'];
43     input1 = int(input("input number, 1: star, 2: square, 3: triangle\n"))
44
45     if input1 ==1:
46         star(t,200,colors[2],colors[0]);
47     elif input1 ==2:
48         square(t,200,colors[3],colors[0]);
49     elif input1 ==3:
50         triangle(t,200,colors[4],colors[0]);
51     t.hideturtle();
52     time.sleep(5);
```

