
python
import turtle
turtle.setup(650,350,200,200)
turtle.penup()
turtle.fd(-250)
turtle.pendown()
turtle.pensize(25)
turtle.pencolor("purple")
turtle.seth(-40)
for 1 in range(4):
turtle.circle(40, 80)
turtle.circle(-40, 80)
turtle.fd(40)
turtle.circle(40, 80/2)
turtle.fd(40)
turtle.circle(16, 180)
turtle.fd(40 * 2/3)

Python语言程序设计

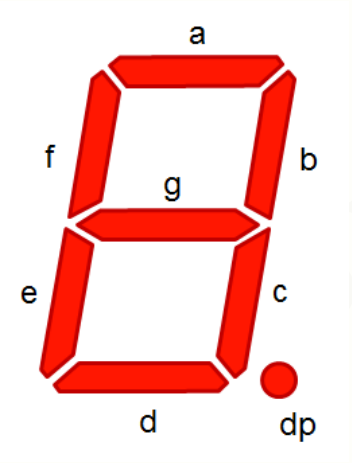
实例7: 七段数码管绘制

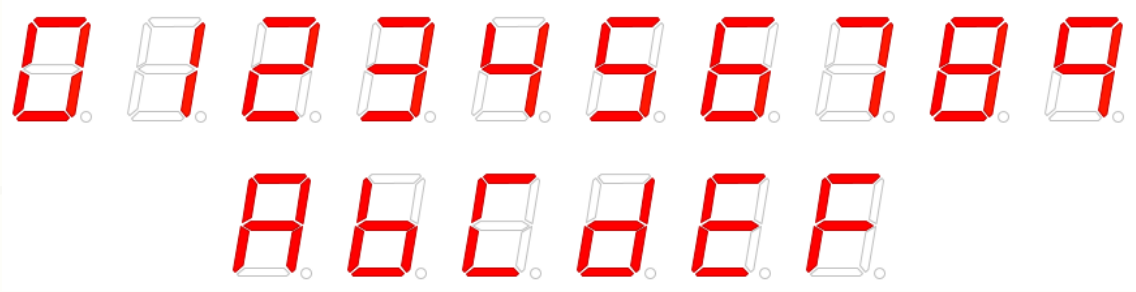
嵩 天
北京理工大学

"七段数码管绘制"问题分析

问题分析

七段数码管





问题分析

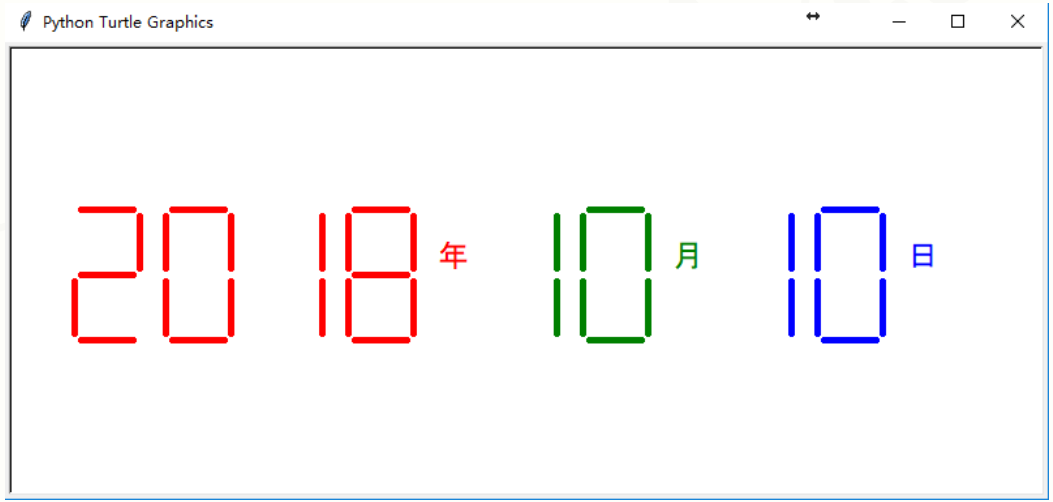
七段数码管绘制

- 需求：用程序绘制七段数码管，似乎很有趣
- 该怎么做呢？

turtle绘图体系 ⇨ 七段数码管绘制

问题分析

七段数码管绘制时间



"七段数码管绘制"实例讲解(上)

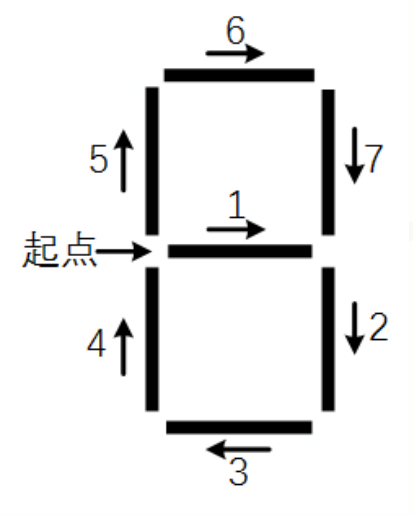
七段数码管绘制

基本思路

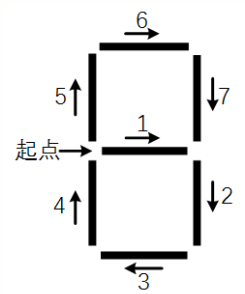
- 步骤1：绘制单个数字对应的数码管
- 步骤2：获得一串数字，绘制对应的数码管
- 步骤3：获得当前系统时间，绘制对应的数码管

七段数码管绘制

步骤1: 绘制单个数码管



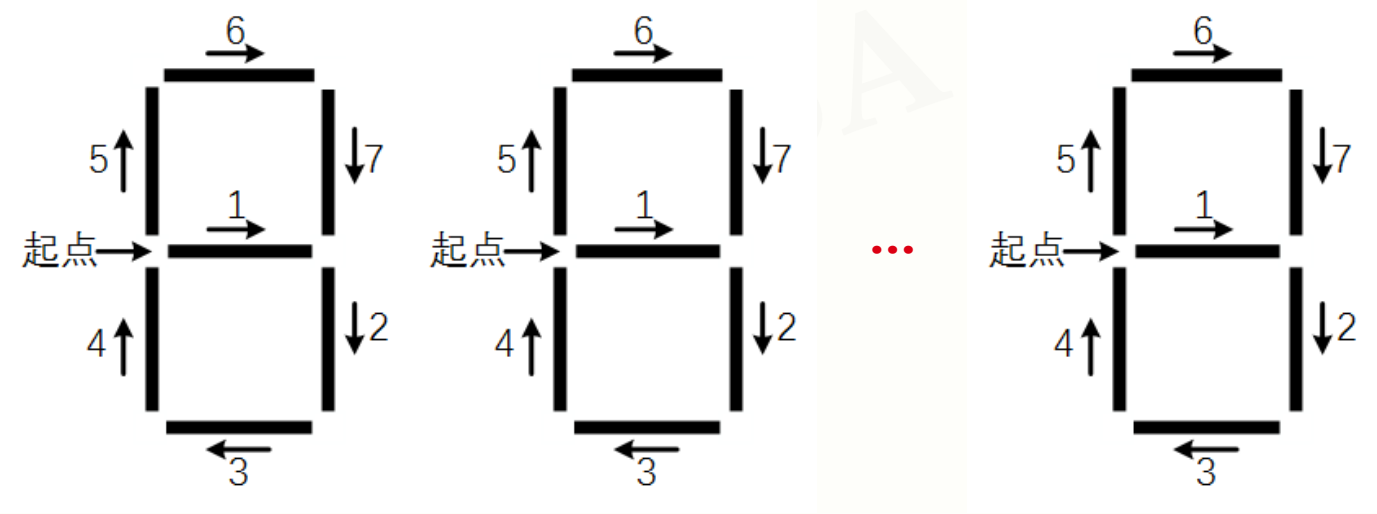
- 七段数码管由7个基本线条组成
- 七段数码管可以有固定顺序
- 不同数字显示不同的线条




```
import turtle
def drawLine(draw):    #绘制单段数码管
    turtle.pendown() if draw else turtle.penup()
    turtle.fd(40)
    turtle.right(90)
def drawDigit(digit): #根据数字绘制七段数码管
    drawLine(True) if digit in [2,3,4,5,6,8,9] else drawLine(False)
    drawLine(True) if digit in [0,1,3,4,5,6,7,8,9] else drawLine(False)
    drawLine(True) if digit in [0,2,3,5,6,8,9] else drawLine(False)
    drawLine(True) if digit in [0,2,6,8] else drawLine(False)
    turtle.left(90)
    drawLine(True) if digit in [0,4,5,6,8,9] else drawLine(False)
    drawLine(True) if digit in [0,2,3,5,6,7,8,9] else drawLine(False)
    drawLine(True) if digit in [0,1,2,3,4,7,8,9] else drawLine(False)
    turtle.left(180)
    turtle.penup() #为绘制后续数字确定位置
    turtle.fd(20)  #为绘制后续数字确定位置
```

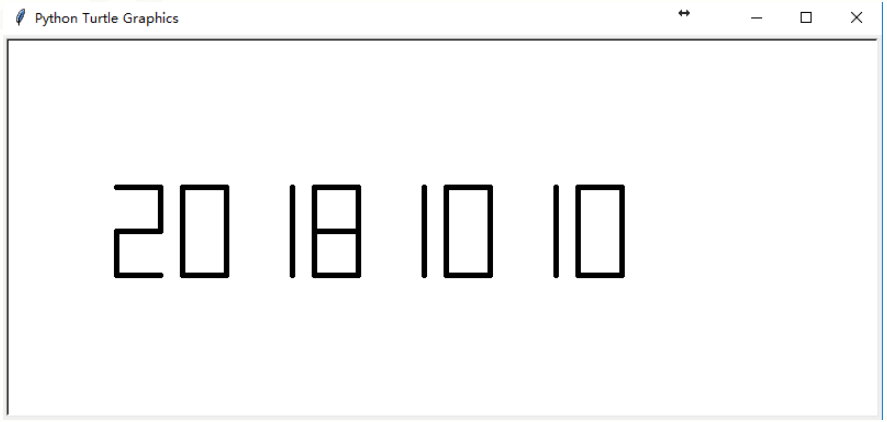
七段数码管绘制

步骤2: 获取一段数字，绘制多个数码管



第1个　　第2个　　第N个

```
import turtle
def drawLine(draw):    #绘制单段数码管
    …(略)
def drawDigit(digit): #根据数字绘制七段数码管
    …(略)
def drawDate(date):    #获得要输出的数字
    for i in date:
        drawDigit(eval(i))  #通过eval()函数将数字变为整数
def main():
    turtle.setup(800, 350, 200, 200)
    turtle.penup()
    turtle.fd(-300)
    turtle.pensize(5)
    drawDate('20181010')
    turtle.hideturtle()
    turtle.done()
main()
```



准备好电脑，与老师一起编码吧！

"七段数码管绘制"实例讲解(下)

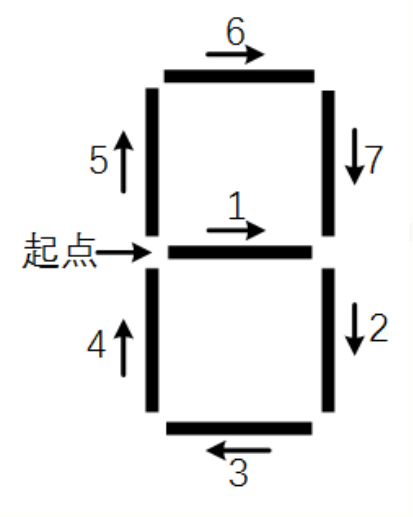
七段数码管绘制

基本思路

- 步骤1：绘制单个数字对应的数码管
- 步骤2：获得一串数字，绘制对应的数码管
- **步骤3：获得当前系统时间，绘制对应的数码管**

七段数码管绘制

绘制漂亮的七段数码管

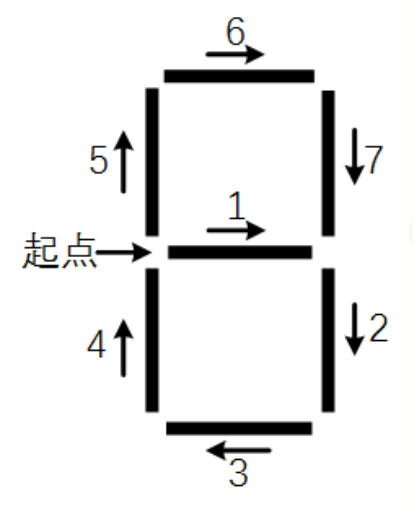


- 增加七段数码管之间线条间隔

```
import turtle
def drawGap():    #绘制数码管间隔
    turtle.penup()
    turtle.fd(5)
def drawLine(draw):    #绘制单段数码管
    drawGap()
    turtle.pendown() if draw else turtle.penup()
    turtle.fd(40)
    drawGap()
    turtle.right(90)
def drawDigit(digit): #根据数字绘制七段数码管
    drawLine(True) if digit in [2,3,4,5,6,8,9] else drawLine(False)
    drawLine(True) if digit in [0,1,3,4,5,6,7,8,9] else drawLine(False)
    drawLine(True) if digit in [0,2,3,5,6,8,9] else drawLine(False)
    drawLine(True) if digit in [0,2,6,8] else drawLine(False)
    …(略)
```

七段数码管绘制

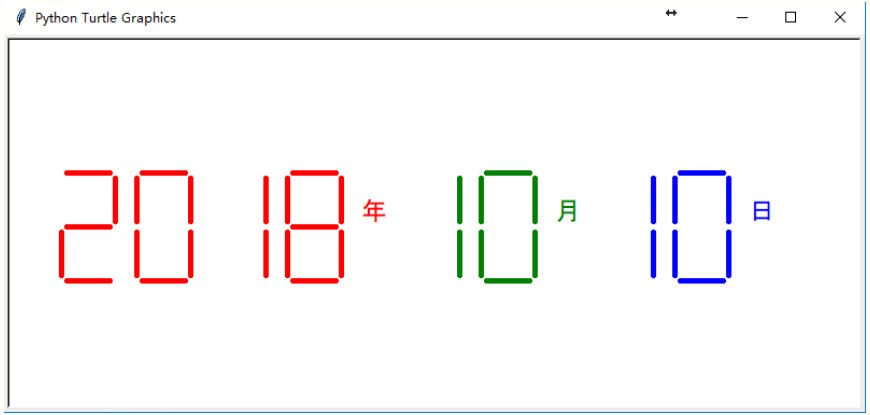
步骤3: 获取系统时间，绘制七段数码管



- 使用time库获得系统当前时间
- 增加年月日标记
- 年月日颜色不同

```
import turtle, time
…(略)
def drawDate(date):   #data为日期，格式为 '%Y-%m=%d+'
    turtle.pencolor("red")
    for i in date:
        if i == '-':
            turtle.write('年',font=("Arial", 18, "normal"))
            turtle.pencolor("green")
            turtle.fd(40)
        elif i == '=':
            turtle.write('月',font=("Arial", 18, "normal"))
            turtle.pencolor("blue")
            turtle.fd(40)
        elif i == '+':
            turtle.write('日',font=("Arial", 18, "normal"))
        else:
            drawDigit(eval(i))
def main():
…(略)
```

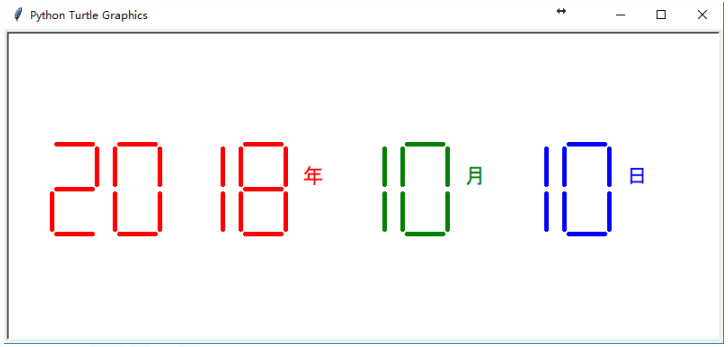
```
import turtle, time
…(略)
def drawDate(date):
…(略)
def main():
    turtle.setup(800, 350, 200, 200)
    turtle.penup()
    turtle.fd(-300)
    turtle.pensize(5)
    drawDate(time.strftime('%Y-%m=%d+',time.gmtime()))
    turtle.hideturtle()
    turtle.done()
main()
```



准备好电脑，与老师一起编码吧！

"七段数码管绘制"举一反三

```
import turtle, time
…(略)
def drawLine(draw):
    drawGap()
    turtle.pendown() if draw else turtle.penup()
    turtle.fd(40)
    drawGap()
    turtle.right(90)
def drawDigit(digit):
    drawLine(True) if digit in [2,3,4,5,6,8,9] else drawLine(False)
    drawLine(True) if digit in [0,1,3,4,5,6,7,8,9] else drawLine(False)
    drawLine(True) if digit in [0,2,3,5,6,8,9] else drawLine(False)
    drawLine(True) if digit in [0,2,6,8] else drawLine(False)
    turtle.left(90)
    drawLine(True) if digit in [0,4,5,6,8,9] else drawLine(False)
    drawLine(True) if digit in [0,2,3,5,6,7,8,9] else drawLine(False)
    drawLine(True) if digit in [0,1,2,3,4,7,8,9] else drawLine(False)
…(略)
```



python

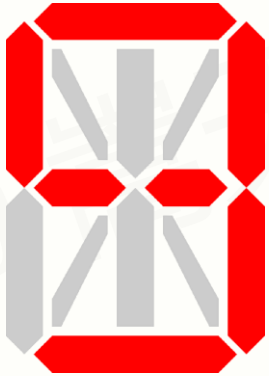
举一反三

理解方法思维

- 模块化思维：确定模块接口，封装功能
- 规则化思维：抽象过程为规则，计算机自动执行
- 化繁为简：将大功能变为小功能组合，分而治之

举一反三

应用问题的扩展



- 绘制带小数点的七段数码管
- 带刷新的时间倒计时效果
- 绘制高级的数码管