计算机实验报告（第2章）

1. 学习目标
2. 掌握解决计算问题的一般方法。
3. 掌握Python语言的基本方法，包括缩进、变量、命名等。
4. 掌握Python语言绘制图形的一般方法。
5. 了解Python标准库的导入和使用。
6. 实验操作

实例1温度转化

根据第1章介绍的程序编写基木方法，用计算机解决上述问题需要6个步骤分析和实现过程如下。

（1）分析问题：可以从很多不同角度来理解旅行者温度转换问题的计算部分。这里给出3个角度。第一，利用程序进行温度转换，由用户输入温度值，程序的输出结果。这是最直观的理解。第二，可以通过语音识别、图像识别等方法自动听并获得温度信息发布渠道（如收音机、电视机等）给出的温度播报源数据，再程序转换后输出给用户。这种角度相比第一种不需要用户给出输入。第三，随着联网的高度普及和接入的便捷，程序也可以定期从温度信息发布网站获得温度值，再将温度信息转换成旅行者熟悉的方式。3 种角度对问题计算部分的不同理解会生成不同的 IPO描述、算法和程序。应该说，“利用计算机解决问题”需要结合计算机技术的发展水平和人类对问题的思考程度，在特定技术和社会条件下，分析出一个问题最经济、最合理的计算部分，进而用程序实现。本文以第一种理解角度编写并讲解余下程序步骤。

（2）划分边界：在确定问题计算部分的基础上进一步划分问题边界，即明确问题的输入数据、输出数据和对数据处理的要求。由于程序可能接收华氏温度 on温度，并相互转换，该功能的IPO描述如下。

输入：带华氏或摄氏标志的温度值

处理：根据温度标志选择适当的温度转换算法

输出：带摄氏或华氏标志的温度值

这里采用82F表示华氏82度，采用28C 表示摄氏28度，实数部分是湿度值。这种温度表示格式同时用于温度的输入和输出。

（3）设计算法：根据华氏和摄氏温度定义，两个温度体系都以1标准大气压下水的结冰点和沸点为温度区间边界，因此，转换算法如下：

C=(F - 32)/1.8

F= C=1.8 + 32

其中，C表示摄氏温度，F表示华氏温度。

（4）编写程序：根据IPO 描述和算法设计，编写如下温度转换的 Python程序

摄氏温 代码：

#el.1TempConvert.py

TempStr = input（"请输入带有符号的温度值："）

if TempStr[-1] in ['F','f']:

C = (eval(TempStr[0:-1])- 32)/1.8

print（"转换后的温度是（:.2fjC".format(C））

elif TempStr[-1] in ['C','c']:

F = 1.8\*eval(TempStr[0:-1]) + 32

print(“转换后的温度是（:.2f）F".format(F））

古着

else:

print（"输入格式错误”）

此时看不懂上述代码没关系，2.2节将逐行解释上述代码的含义。

（5）调试测试：将上述程序保存为文件：el.lTempConvert.py，

输入带华氏标志的温度 的e结果如下：

输入带摄氏标志的温度值，程序运行结果如下：

>>>

请输入带有符号的温度值：-30C

转换后的温度是-22.00F

上述程序符合Python 语法，执行结果正确，事实上，当程序较为复杂时，很难保证一次写后的程序能够直接正确运行或运行逻辑没有错误。甚至说，任何程序都会有错误。寻找错误的调试过程不容忽视。

（6）升级维护：与人一样，任何程序都有生命周期。促使程序生命结束的事件有很多，例如，平台更换、使用方式变化、算法改进等。对于上述例子，只要中国、美国使用不同的温度标准，温度转换问题将一直存在。随着问题使用场景、输入和输出要求等因素的变化，程序将需要不断地维护和升级。

实例2：Python 蛇绘制

Python 英文是“蝴蛇”的意思，因此，绘制一条鳞蛇十分有趣。本节以“Pytho

締蛇绘制”为例，介绍使用Python 绘制图形程序的基本方法，并讲解Python语

的“模块编程”思想。

实例代码2.1是“Python 嫌蛇绘制”的源代码，

实例代码 2.1

#e2.1DrawPython.py

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 i in range (4) ;

turtle.circle(-40,80) turtle.circle(40,80)

turtle.circle(40，80/2)

turtle.fd(40)

turtle.circlel(16,180)

turtle.fd(40\*2/3)

实例代码2.1 与实例代码 1.1有两个显者的不同。第一，这个程序没有使用显 式的用户输入输出，即没有 inputO函数和 print(）函数；第二，这个程序绝大多数代码行都是<a>.<b>0形式，代码行中没有赋值语句。

<a>.<b>0是 Python 编程的--种典型表达形式，它可以表示调用一个对象"<a>的方法<b>0，也可以表示调用一个函数库<a>中的函数<b>0。

实例代码2.1使用了用于绘制图形的 turtle库，并在第2行代码中通过保留字import 引用这个函数库。

import turtle

实例代码2.1的第3行到第16行调用了turtle库中若干函数来绘制Python蛇，

所有被调用的函数都使用了<a>.<b>0形式。这种通过使用函数库并利用库中函数进

行编程的方法是 Python 语言最重要的特点，称为“模块编程”。8.5节将详细介绍

Python 模块编程思想以及本书所提出的面向“计算生态”的教学理念。

使用import引用函涵数库有两种方式，但对函数的使用方式略有不

第一种引用函数库的方法如下：

import <库名>

此时，程序可以调用库名中的所有函数，使用库中函数的格式如下

<库名>.<函数名>(<函数参数>)

第二种引用函数库的方法如下：

from <库名> import<函数名，函数名，··，函数名> f其中，·是通配符，表示所有函数

from <库名> import \*

此时，调用该库的函数时不再需要使用库名，直接使用如下格式：

<函数名>(<函数参数>)

采用第二种库引用方式修改实例代码2.2完成 Python 娇蛇绘制，代码如

实例代码 2.2

#e2.2DrawPython.py

from turtle import \*

setup(650，350，200,200)

penup ()

fd(-250)

pendown()

pensize(25)

pencolor（"puzple")

seth(-40)

for i in range (4):

circle(40,80)

circle(-40,80)

circle(40，80/2)

fd(40)

circle(16, 180)

fd(40•2/3)

实例代码 2.2 与实例代码 2.1 运行结果相同，所不同的是调用 turtle 库中函数时不再采用<a>.<b>0方式，而直接使用函数名。由于“Python蛇绘制”程序只用了turtle 库中的 sctup(、penup(、fdO、pendown(、pensizeO、pencolor、seth()、circle(等8个函数，第2行的 import 语句也可以写成如下形式：

3 from turtle import setup, penup, fd, pendownfrom turtle import pensize, pencolor, seth, circle

两种函数库引用方式各有优点。第一种采用<a>.<b>0方式调用库中函数，能够显式标明函数来源，在引用较多库时代码可读性更好。第二种利用保留字直接引用库中函数，可以使代码更简洁，在类似实例代码2.2这种只引用一个库的情况下，效果更好。

需要注意的是，第一种引用方式，Python 解释器将<a>.<b>整体作为函数名。 当采用第二种方式时，Python 解释器将<b>作为函数名。这可能产生一种情况，假设用户已经定义了一个函数<b>，库中的函数名<b>将会与用户自定义的函数名冲突。由于 Python 程序要求函数命名唯一，所以，当函数名冲突时 Python 解释器会以最近的函数定义为准。为了避免可能的命名冲突，对于初学者，建议采用第一种库引用方式，使用<a>.<b>0方式调用库函数。

turtle 库语法元素分析

turtle 库绘制图形有一个基本框架：一个小海龟在坐标系中爬行，其爬行轨迹形成了绘制图形。对于小海龟来说，有“前进”、“后退”、“旋转”等爬行行为，对坐标系的探索也通过“前进方向”、“后退方向”、“左侧方向”和“右侧方向”等小海龟自身角度方位来完成。刚开始绘制时，小海龟位于画布正中央，此处坐标为（00），行进方向为水平右方。例如，用如下代码绘制如图2.4所示的图坐标体系。

turtle.setup(650，350，200，200)

实例代码2.1第3行使用了 turtle.setup()函数，该函数各参数的关系如图所示，其具体定义如下。

tortle.setup(width, height, startx, starty)

作用：设置主窗体的大小和位置。

参数如下。

width: 窗口宽度，如果值是整数，表示像素值；如果值是小数，表示窗口宽度

与屏幕的比例。

height：窗口高度，如果值是整数， 表示像素值；如果值是小数，表示窗口高

度与屏幕的比例。

startx：窗口左侧与屏幕左侧的像素距离，如果值是 None，窗口位于屏幕水平中央

starty: 窗口项部与屏幕顶部的像素距离，如果值是 None，窗口位于屏幕垂直中央。

2.4.2 画笔控制函数

1. turtle.pemup()和 turtle.pendown()函数

turtle.penup()

turtle.pendown () (实例代码 2.1 中第4、6行）

turtle 中的画笔（即小海龟）可以通过一组函数来控制，实例代码2.1中第4行

的 turtlc.:penup()函数和第6 行的 turtle.pendown()函数是一组，它们分别表示抬起画

笔和落下画笔，函数定义如下;

turtle.penup()

别名

turtle.pu(), turtle.up()

作用：抬起画笔，之后移动画笔将绘制形状。

参数：无。

turtle.pendown()

别名

turtle.pd(),turtle.down()

作用：落下画笔，之后移动画笔将绘制形状。

参数：无。

2. turtle.pensize()函数

turtle.pensize(25） （实例代码 2.1 中第7行）

turtle.pensizc()函数用来设置画笔尺寸，函数定义如下：

turtle.pensize(width)

别名

turtle.width()

作用：设置画笔宽度，当无参数输入时返回当前画笔宽度。

参数如下。

width：设置的画笔线条宽度，如果为None或者为空，则函数返回当前画笔宽度。

3.turtle.pencolor0函数

turtle.pencolor（实例代码2.1中第8行）

turtle.pencolorC)函数给画笔设置颜色，实例代码2.1中将画笔设为“紫色”，函数定义如下：

turtie.pencelor(colorstring)

或

turtle.peneolor((r.g.b)) 成

作用：设置画笔颜色. 当无参数输入时返回当前画笔颜色。

参数如下。

colorstring ：表示颜色的字符串，例如，“purple"、"red"、"blue"等，

（5g.b）：颜色对应的 RGB 数值，例如， (51，204，140）.

很多RGB 液色器看周定的黃文名字，这些英文名字可以作为colorslring输入新

ohe Ponoe網景书世奇以采用.B.b）形式直接输入颜色值。几种典型的RGB

颜色如表2.2所示。

2.4.3 形状绘制函数

tunle.fdO函数

turtle.fd(-250)

turtle.fd(40) (实例代码 2.1 中第5、14、16行）

turtle.fd(40·2/3)

turtle 通过一组函数控制画笔的行进动作，进而绘制形状。turtle.fdO)函数最常用

来控制画笔向当前行进方向前进一个距离，函数定义如下:

turtle.fd(distance)

别名

turtle.forward(distance)

作用：向小海龟当前行进方向前进 distance 距离.

参数如下。

distance: 行进距离的像素值，当值为负数时，表示向相反方向前进。

实例代码 2.1 中第 5、14、16 行分别表示向画笔当前前进方向或反方向行进一

段距离，进而绘制一条直线。第5行代码由于配合了第4行抬起画笔函数，因此，

它将不绘制一条直线，而是将画笔移动到某个位置。

2. turtle.seth()函数

turtle.seth(-40)

turtle.seth()函数用来改变画笔绘制方向，函数定义如下：

turtle.seth(to\_angle)

别名

turtle.setheading(to\_angle)

作用：设置小海龟当前行进方向为to\_angle，该角度是绝对方向角度值。

参数如下。

to\_angle：角度的整数值。

如图2.6所示是 turtle库的角度坐标体系，供turtle.seth()等函数使用。需要注意

的是，turtle库的角度坐标体系以正东向为绝对0度。这也是小海龟初始爬行方向，

正西方向为绝对180度，这个方向是方向的绝对方向体系，与小海龟爬向当

前方向无关。因此，可以利用这个绝对坐标体系随时更改小海龟的前进方向。 实例代码2.1中第9行将小乌龟的前进方向设定为-40度，即320度，之后小海龟向这个方向前进。 3、for 循环语句和 turtle.ircle(函数

tor s in range(4) i (实例代码 2.1 中第 10~15行）

turtle.cirele(40,80)

turtle.cirele(-40,80)

12 turtie.cirele(40,80/2)

turtie.circle(16,180)

由于存在缩进，实例代码2.中第10、11、12行是一个由保留字 for 引导的整

体，这是另一种循环结构，称为“遍历循环”。for 语句的循环格式如下：

实例代码 for l in range(<循环次数>) :

《语句块 1>

2.1 中第 10行的 for 循环表示第 11、12行代码连续执行4次。

turle cirde0函数用来差制一-个張形，函数定义如下

作用：根据半径radius 绘制extent 角度的弧形，

参数如下。

radius：弧形半径，当值为正数时，半径在小海龟左侧，当值为负数时，半径在小海龟右侧。

extent：绘制弧形的角度，当不设置参数或参数设置为None时，绘制整个圆形。

实例代码 2.1 中各函数包括角度值、半径值等参数是根据绘制内容的样式调整确定的，在了解各函数使用含义的基础上，读者可以修改各函数参考值，观察蟒蛇绘制的效果。

1. 本章小结

本章从解决实际问题入手，以简单的温度转换程序为例，逐行逐句地分析了

Python语言的基本元素。通过讲解Python 蜡蛇绘制实例，介绍Python语言函数库

turtle 及其基本用法。