**2.4可以复用的代码**

**——火柴棒摆数字**

**【学科核心素养】**

1. 能够根据解决问题的需要，自觉、主动地寻求恰当的方式处理信息。（信息意识）
2. 通过判断、分析与综合各种信息资源，运用合理的算法形成解决问题的方案。（计算思维）
3. 掌握数字化学习工具的操作技能。（数字化学习与创新）
4. 能够遵守信息法律法规，信守信息社会的道德与伦理准则。（信息社会责任）

**【课程标准要求】**

1.7掌握一种程序设计语言的基本知识，使用程序设计语言实现简单算法。通过解决实际问题，体验程序设计的基本流程，感受算法的效率，掌握程序调试与运行的方法。

**【学业要求】**

依据解决问题的需要，设计和表示简单算法；掌握一种程序设计语言的基本知识，利用程序设计语言实现简单算法，解决实际问题（计算思维）。

**【学情分析】**

学生在前面的课中已经学习了基本的赋值语句、分支语句和循环语句，对于编写程序解决问题的一般流程已经比较熟悉，但学生对函数和代码发布应该没有基础。

**【学习目标】**

1.了解函数的作用。

2.掌握函数的定义和调用方法。

3.了解发布共享代码的常见方法。

**【教学重点】**

1.函数的定义及调用。

2.模块的发布和导入使用。

**【教学难点】**

共享代码的发布。

**【教学过程】**

**第一课时**

一、引入

有6根火柴棒，列出所有能摆出的自然数（要求火柴棒正好用完）。



图2.4.1 火柴数字

问题1. 列举3个可以用6根火柴棒摆放的数字\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_。

2. 6根火柴棒能摆放的最大数是\_\_\_\_\_\_\_\_\_\_\_\_。

参考答案：1. 0、6、9、14、41、77、111

2. 111

二、函数与调用

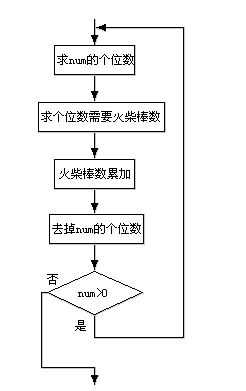
1. 算法分析

* 数字“1”需要的火柴棒最少，因此，能摆出的自然数最大为111。
* 对于0～111之间的每一个数进行计算，统计需要的火柴棒个数。
* 凡是恰好需要6根火柴棒的数据直接打印输出。

2.用流程图表示

将这个数的每个数位上的数字所需要的火柴棒数加起来。实现方法是将这个数除以10取余，求得个位数；然后将这个个位数需要的火柴棒数累加；接着将这个数除以10取整，去掉个位数；重复以上步骤直到这个数为0。思考：如果一开始num就等于0，怎么办？

参考答案：



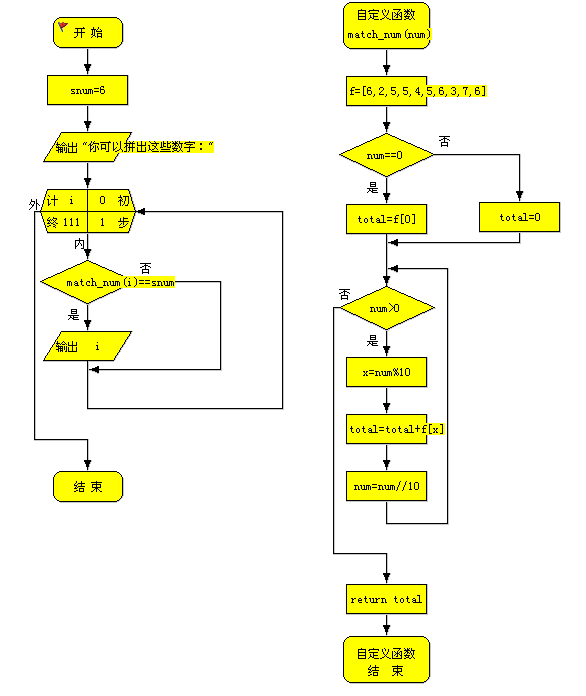
根据以上分析，对每一个自然数进行火柴棒个数的统计需要重复使用，因此可以设计函数优化代码。

（3）画程实现

计算自然数num需要多少根火柴棒，可以分别计算num中各个数位需要多少根火柴，然后将各数位对应的火柴棒根数进行累加。

如何获得num各数位的值？➀将num除以10取余，求得个位数；➁然后将num除以10取整，去掉个位数。重复步骤➀➁直到num为0。显然可用while循环。

单个数字0～9需要的火柴棒根数可预先存储到列表f中。



**第二课时**

1. 函数的定义

函数分类：系统函数和用户自定义函数，体现共性和个性的问题。

自定义函数：

函数一般包括函数名、参数、返回值和函数体4部分。其中函数名和函数体是必不可少的，参数和返回值可根据需要进行定义。定义函数的基本格式是：

def 函数名(参数):

# 函数说明

语句或语句组

return 返回值

参数是用来向函数传递值的，当有多个参数时各个参数由逗号分隔。函数执行完成后，由return语句将表达式值返回给调用者，结束函数。

1. 将画程程序转换为Python程序

def match\_num(num):

f=[6,2,5,5,4,5,6,3,7,6] # 0～9的数字分别需要多少根小棒

if num==0:

total=6

else:

total=0 # 火柴棒总数变量赋初值

while (num>0):

x=num % 10 # 取num除以10的余数，即num的个位数

total=total+f[x] # 所需火柴棒数累加

num=num//10 #num整除10，即去掉num的个位数

return total #返回需要多少根火柴棒数

#以下为主程序

snum=6 # 6根火柴棒

print("你可以拼出这些数字：")

for i in range(112):

if match\_num(i)==snum: #如果i需要的火柴棒数等于现有火柴棒数

print (i)

注意教材中的写法：total=0 if num!=0 else f[0]

三、发布及导入、使用模块

已经完成上面程序编写的同学，2人一组，阅读下面的资料，仿照相关步骤进行。

A同学生成相应模块并发布，将发布后的文件夹共享给B。

B同学将发布后的共享文件夹下载并在本地安装，导入模块并使用，实现火柴棒摆数字问题。

**（一）发布代码**

首先做好发布前的准备工作。在D:\创建一个文件夹mymodule，其中包含mymodule.py和setup.py（包含发布的元数据）两个文件。文件内容如下：

mymodule.py是一个模块文件，其内容就是match\_num函数。

def match\_num(num):

f=[6,2,5,5,4,5,6,3,7,6] # 0～9的数字分别需要多少根火柴棒

if num==0:

total=6

else:

total=0 # 火柴棒总数变量赋初值

while (num>0):

x=num % 10 # 取num除以10的余数，即num的个位数

total=total+f[x] # 所需火柴棒数累加

num=num//10 #num整除10，即去掉num的个位数

return total #返回需要多少根火柴棒数

setup.py文件包含有关发布的元数据，代码如下。

from distutils.core import setup

setup(

name ='mymodule', #参数一定要与mymodule.py文件名相同

version ='1.0.0', #版本号

py\_modules =['mymodule'], #参数一定要与mymodule.py文件名相同

author='yd', #作者

author\_email='yd@mail.com', #作者邮箱

description ='计算一个数字需要多少根火柴棒', #代码功能描述

)

在DOS命令行中打开mymodule所在的目录，构建发布文件，命令如图1。请注意本机Python安装的位置，例子中Python安装在d:\python34文件夹中。

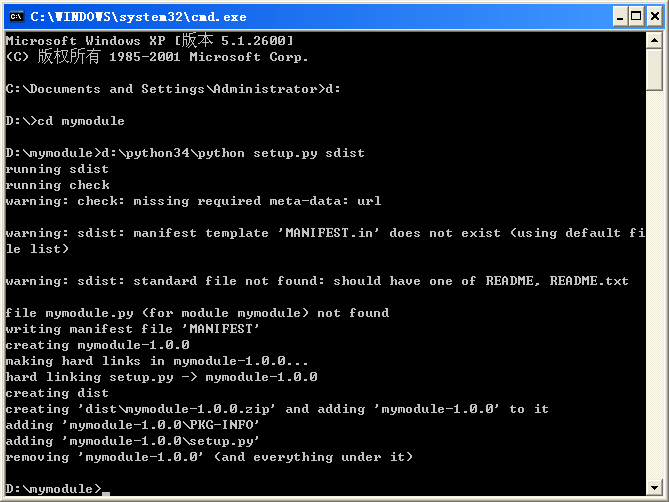


图1 构建发布文件

最后将发布安装到你的Python本地副本中，如图2。

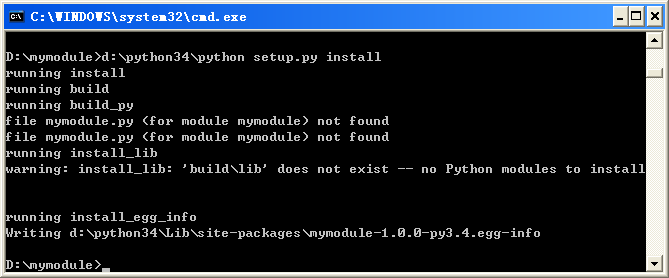


图2 安装发布

安装完成后mymodule文件夹包含了如图3的文件。

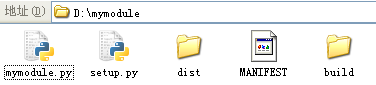
****

图3 mymodule文件夹内容

如果你想在因特网上发布代码，让全世界都分享你的成果。可以访问PyPI网站（访问这个网站速度很慢）注册一个PyPI ID，然后向PyPI上传代码。PyPI(Python Package Index)是python官方的第三方库的仓库，所有人都可以下载第三方库或上传自己开发的库到PyPI。

**（二）导入模块并使用**

已经构建发布的模块必须先导入程序，然后才能使用。导入时使用语句“import mymodule”，调用模块内的函数时用“mymodule.match\_num()”来实现。

import mymodule #导入模块

snum=6

print("你可以拼出这些数字：")

for i in range(112):

if mymodule.match\_num() ==snum: #调用模块内match\_num()函数

print (i)

四、小结

函数的作用；函数的定义和调用；共享代码的发布。