程序1： 数字组合 2

程序2：猜猜这个数是多少？ 3

程序3：判断这是一年中的第几天？ 4

程序4：判断整数大小 5

程序5：斐波那契数列 6

程序6：水仙花数 7

程序7：数字求和 8

程序8：平方根 9

程序9：if语句 10

程序10：阿姆斯特朗数 11

程序11：输出指定范围的素数（else语句） 12

程序12：生成日历 14

程序13：统计字符串中的字符 15

程序14：文件的读取 16

程序15：十进制转二进制、八进制、十六进制 17

程序16： 生成10个两位的随机素食 18

程序17： 计算三角形面积 20

程序18：字符串大小写转换 21

程序19： 获取昨天日期 22

程序20： 变量交换 22

程序21： 质数判断 23

程序22： 简单计算器 24

程序23： 最大公约数 27

程序24： 最小公倍数 28

程序25： 字符串判断 29

程序26： 合并文件数据 30

程序27： 猜数游戏 31

程序28：为数据加密 33

程序29：平方运算 34

程序30： 计算0-7组成的奇数个数 35

程序31：求值 36

程序32：猴子分桃 37

程序33：淘汰游戏 38

程序34：分解质因子 39

程序35：取整数右端的4~7位 40

程序36：判断4位回文数 41

程序37：汉诺塔（递归函数） 42

程序38：判断闰年 44

程序39：寻找自幂数（38） 45

程序40：多维数据 47

程序41： 验证哥德巴赫猜想 48

程序42： 解方程 49

程序43： 异常处理 51

程序44： 统计不及格人数（continue语句） 52

程序45： 求直角三角形的斜边 52

程序46： 问年龄 54

程序47：求相同数字的和 54

程序48：找出1000以内的所有完数 56

程序49：公民类 56

程序50：复数类 58

## 程序1：数字组合

#### 【程序描述】

有四个数字：1、2、3、4，能组成多少个互不相同且无重复数字的三位数？各是多少？

#### 【程序分析】

可填在百位、十位、个位的数字都是1、2、3、4。组成所有的排列后再去掉不满足条件的排列。

#### 【程序实现】

for i in range(1,5):  
 for j in range(1,5):  
 for k in range(1,5):  
 if i!=j and j!=k and i!=k:  
 print(i,j,k)

#### 【程序结果】

1 2 3

1 2 4

1 3 2

1 3 4

1 4 2

1 4 3

2 1 3

2 1 4

2 3 1

2 3 4

2 4 1

2 4 3

3 1 2

3 1 4

3 2 1

3 2 4

3 4 1

3 4 2

4 1 2

4 1 3

4 2 1

4 2 3

4 3 1

4 3 2

## 程序2：猜猜这个数是多少？

#### 【程序描述】

一个整数，它加上100和加上268后都是一个完全平方数，请问该数是多少？

#### 【程序分析】

在10000以内判断，将该数i加上100后再开方，加上268后再开方，如果开方后的结果满足如下条件，即是结果。

xx =（i+100）

yy=（i+268）

#### 【程序实现】

import math

for i in range(1,10000):  
 x=int(math.sqrt(i+100))  
 y=int(math.sqrt(i+268))  
 if (x\*x==i+100)and(y\*y==i+268):  
 print(i)

#### 【程序结果】

21

261

1581

## 程序3：判断这是一年中的第几天？

#### 【程序描述】

输入某年某月某日，判断这一天是这一年的第几天？

#### 【程序分析】

以3月5日为例，应该先把前两个月的加起来，然后再加上5天即本年的第几天，特殊情况，闰年且输入月份大于2时需考虑多加一天。

#### 【程序实现】

year=int(input("请输入年份"))  
 month=int(input("请输入月份"))  
 day=int(input("请输入日期"))  
 months=(0,31,59,90,120,151,181,212,243,273,304,334)  
 if 0<month<=12:  
 sum=months[month-1]  
 else:  
 print('输入的月份有误')  
 sum+=day  
 leap=0  
 if (year%400==0) or ((year%4==0) and (year%100!=0)):  
 leap=1  
 if(leap==1)and(month>2):  
 sum+=1  
 print("今天是第%d天"%sum)

#### 【程序结果】

请输入年份2000

请输入月份3

请输入日期1

今天是第61天

## 程序4：判断整数大小

#### 【程序描述】

输入三个整数x,y,z，请把这三个数由小到大输出。

#### 【程序分析】

可以借助列表的sort方法完成排序。首先将用户输入的整数存放在列表中，列表从小到大排序后，通过遍历的方式输出列表中的值即可。

#### 【程序实现】

x=int(input("请输入进行比较的第1个数："))  
 y=int(input("请输入进行比较的第2个数："))  
 z=int(input("请输入进行比较的第3个数："))

list=[x,y,z]  
 list.sort()  
 for i in range(len(list)):  
 print(list[i])

#### 【程序结果】

请输入进行比较的第1个数：34

请输入进行比较的第2个数：23

请输入进行比较的第3个数：55

23

34

55

## 程序5：斐波那契数列

#### 【程序描述】

编写一个程序，实现斐波那契数列。

#### 【程序分析】

斐波那契数列（Fibonacci sequence），又称黄金分割数列，指的是这样一个数列：0、1、1、2、3、5、8、13、21、34、……。

在数学上，费波那契数列是以递归的方法来定义：

F0 = 0 (n=0)

F1 = 1 (n=1)

Fn = F[n-1]+ F[n-2](n=>2)

#### 【程序实现】

def fib(n):  
 if n==1 or n==2:  
 return 1  
 return fib(n-1)+fib(n-2)

print(fib(10))

#### 【程序结果】

55

## 程序6：水仙花数

#### 【程序描述】

编写程序，打印出所有的“水仙花数” 。

#### 【程序分析】

所谓“水仙花数”是指一个三位数，其各位数字立方和等于该数本身。例如：153是一个“水仙花数”，因为153=1的三次方＋5的三次方＋3的三次方。

#### 【程序实现】

for i in range(100,1000):  
 a=int(i/100)  
 b=int(i/10)%10  
 c=i%10  
 if i==a\*\*3+b\*\*3+c\*\*3:  
 print(i)

#### 【程序结果】

153

370

371

407

## 程序7：数字求和

#### 【程序描述】

编写一个程序，通过用户输入两个数字，并计算这两个数字之和。

#### 【程序分析】

根据程序描述，该程序实现过程大概分为如下几步：

1. 提示用户输入第1个数字
2. 提示用户输入第2个数字
3. 对两个数字进行相加
4. 输出两个数字相加的和

#### 【程序实现】

num1=input("请输入第1个数字：")  
 num2=input("请输入第2个数字：")  
 sum=float(num1)+float(num2)  
 print("数字%s和%s相加之和是%.1f"%(num1,num2,sum))

#### 【程序结果】

请输入第1个数字：12.3

请输入第2个数字：10.4

数字12.3和10.4相加之和是22.7

## 程序8：平方根

#### 【程序描述】

平方根，又叫二次方根，表示为〔√￣〕，如：数学语言为：√￣16=4。语言描述为：根号下16=4。编写一个程序，通过用户输入一个数字，并计算这个数字的平方根。

#### 【程序分析】

关于计算平方根的，可以使用在指数运算符 \*\* 来计算改数的平方根，也可以使用math函数中的sqrt，不同的是，指数\*\*只适用于正数，而sqrt适用于任何数。接下来，按照下列步骤实现程序：

1. 提示用户输入一个数(必须将输入的数转为数字类型）
2. 使用math函数的sqrt计算这个数的平方根
3. 输出运算结果

#### 【程序实现】

import math  
 num=int(input("请输入要计算平方根的数字："))  
 result=math.sqrt(num)  
 print("数字%d的平方根是%f"%(num,result))

#### 【程序结果】

请输入要计算平方根的数字：12

数字12的平方根是3.464102

## 程序9：if语句

#### 【程序描述】

编写程序，通过使用 if...elif...else 语句判断数字是正数、负数或零：

#### 【程序分析】

正数、负数或零的判断非常简单，只需要判断这个数是否大于零，小于零或者等于零。由于判断的条件大于2个，这里我们使用if…elif…else判断。

#### 【程序实现】

num = float(input("输入一个数字: "))  
 if num > 0:  
 print("正数")  
 elif num == 0:  
 print("零")  
 else:  
 print("负数")

#### 【程序结果】

输入一个数字: -2

负数

## 程序10：阿姆斯特朗数

#### 【程序描述】

如果一个n位正整数等于其各位数字的n次方之和,则称该数为阿姆斯特朗数。 例如1^3 + 5^3 + 3^3 = 153。

1000以内的阿姆斯特朗数： 1, 2, 3, 4, 5, 6, 7, 8, 9, 153, 370, 371, 407。

请编写程序，检测用户输入的数字是否为阿姆斯特朗数。

#### 【程序分析】

#### 【程序实现】

# 获取用户输入的数字

num = int(input("请输入一个数字: "))

# 初始化变量 sum

sum = 0

# 指数

n = len(str(num))

# 检测

temp = num

while temp > 0:

digit = temp % 10

sum += digit \*\* n

temp //= 10

# 输出结果

if num == sum:

print(num,"是阿姆斯特朗数")

else:

print(num,"不是阿姆斯特朗数")

#### 【程序结果】

请输入一个数字: 12

12 不是阿姆斯特朗数

或者

请输入一个数字: 153

153 是阿姆斯特朗数

## 程序11：输出指定范围的素数（else语句）

#### 【程序描述】

素数（prime number）又称质数，有无限个。除了1和它本身以外不再被其他的除数整除。请编写一个程序，输出指定范围内的素数：

#### 【程序分析】

由程序描述可知，要想输出指定范围的素数，可以按照下列步骤开发：

1. 提示用户输入指定的范围，分别是范围的最小值和最大值
2. 使用for循环在指定范围内进行循环
3. 由于满足素数的条件是只能被1和本身整除，所以可以使用if语句判断是否能整除，如果能，则输出素数。

#### 【程序实现】

lower = int(input("输入区间最小值: "))

upper = int(input("输入区间最大值: "))

for num in range(lower,upper + 1):

# 素数大于 1

if num > 1:

for i in range(2,num):

if (num % i) == 0:

break

else:

print(num)

#### 【程序结果】

输入区间最小值: 2

输入区间最大值: 50

2

3

5

7

11

13

17

19

23

29

31

37

41

43

47

## 程序12：生成日历

#### 【程序描述】

请编写程序，用于生成指定日期的日历。

#### 【程序分析】

Calendar模块有很广泛的方法用来处理年历和月历，例如打印某月的月历。

#### 【程序实现】

# 引入日历模块  
 import calendar  
 # 输入指定年月  
 yy = int(input("输入年份: "))  
 mm = int(input("输入月份: "))  
 # 显示日历  
 print(calendar.month(yy,mm))

#### 【程序结果】

输入年份: 2017

输入月份: 3

March 2017

Mo Tu We Th Fr Sa Su

1 2 3 4 5

6 7 8 9 10 11 12

13 14 15 16 17 18 19

20 21 22 23 24 25 26

27 28 29 30 31

## 程序13：统计字符串中的字符

#### 【程序描述】

输入一行字符，分别统计出其中英文字母、空格、数字和其它字符的个数。

#### 【程序分析】

开发步骤如下：

1. 提示用户输入字符串
2. 声明四个变量，分别用于统计英文字母、空格、数字和其他字符的个数
3. 使用for循环遍历字符串中的字符，使用if判断语句判断遍历的字符满足哪个条件，如果满足条件，则对应的变量加1

#### 【程序实现】

str=input("请输入一个字符串:")  
 letters = 0  
 space = 0  
 digit = 0  
 others = 0  
 for c in str:  
 if c.isalpha():  
 letters += 1  
 elif c.isspace():  
 space += 1  
 elif c.isdigit():  
 digit += 1  
 else:  
 others += 1  
 print("字符串共有%d个英文字母，%d个空格，%d个数字，%d个其他字符

"%(letters,space,digit,others))

#### 【程序结果】

请输入一个字符串:hello world,i love you!

字符串共有18个英文字母，3个空格，0个数字，2个其他字符

## 程序14：文件的读取

#### 【程序描述】

从键盘输入一个字符串，将小写字母全部转换成大写字母，然后输出到一个磁盘文件"test"中保存。

#### 【程序分析】

开发步骤如下：

1. 提示用户输入一个字符串
2. 使用字符串的upper方法将字符串的小写字母全部转换为大写字母
3. 使用write方法将字符串写入文件，需要注意的是，在写文件之前，首先要打开文件，完成写数据的操作后，要切记关闭文件。

#### 【程序实现】

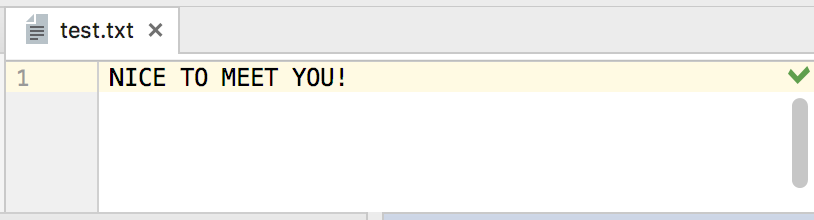
fp = open('test.txt','w')  
 string = input('please input a string:\n')  
 string = string.upper()  
 fp.write(string)  
 fp = open('test.txt','r')  
 print(fp.read())  
 fp.close()

#### 【程序输出】

please input a string:

Nice to meet you!

NICE TO MEET YOU!



## 程序15：十进制转二进制、八进制、十六进制

#### 【程序描述】

编写程序，用于实现十进制转二进制、八进制、十六进制。

#### 【程序分析】

不同进制之间的转换，需要用到下列方法：

1. bin()：将十进制的数转为二进制
2. oct()：将十进制的数转为八进制
3. hex(): 将十进制的数转为十六进制

#### 【程序实现】

# 获取用户输入十进制数

dec = int(input("输入数字："))

print("十进制数为：", dec)

print("转换为二进制为：", bin(dec))

print("转换为八进制为：", oct(dec))

print("转换为十六进制为：", hex(dec))

#### 【程序结果】

输入数字：123

十进制数为： 123

转换为二进制为： 0b1111011

转换为八进制为： 0o173

转换为十六进制为： 0x7b

## 程序16： 生成10个两位的随机素食

#### 【程序描述】

请编写一个程序，输出生成10个两位的随机数，并且这两个随机数是素数。

#### 【程序分析】

Python的random模块提供了随机数生成方法random.randint(a,b),它返回一个大于等于a，小于等于b的随机整数。

#### 【程序实现】

import random  
 n=0  
 while n<10:  
 x=random.randint(10,99) #获得一个两位的随机整数  
 #判断x是否为素数  
 a=2  
 while a<x-1:  
 if x%a==0: #若余数为0，说明x不是素数，结束当前循环  
 break  
 a+=1  
 else:  
 print(x) #若正常结束循环时，说明x是素数，输出

n+=1 #累计素数个数

#### 【程序结果】（每次结果不一样）

61

29

83

41

73

97

19

11

31

17

## 程序17： 计算三角形面积

#### 【程序描述】

通过用户输入三角形三边长度，并计算三角形的面积。

#### 【程序分析】

要想计算三角形的面积，需要获取三角形的三边长度，假设三角形的三边分别是a，b，c，那么面积的计算方式如下：

三角形半周长s=(a+b+c)/2

三角形面积=(s\*(s-a)\*(s-b)\*(s-c)) \*\* 0.5

#### 【程序实现】

a = float(input('输入三角形第一边长: '))  
 b = float(input('输入三角形第二边长: '))  
 c = float(input('输入三角形第三边长: '))  
 # 计算半周长  
 s = (a + b + c) / 2  
 # 计算面积  
 area = (s\*(s-a)\*(s-b)\*(s-c)) \*\* 0.5  
 print('三角形面积为 %0.2f' %area)

#### 【程序结果】

输入三角形第一边长: 12

输入三角形第二边长: 23

输入三角形第三边长: 21

三角形面积为 125.22

## 程序18：字符串大小写转换

#### 【程序描述】

如何将字符串转换为大写字母，或者将字符串转为小写字母等。

#### 【程序分析】

字符串str提供了很多转换字母的方法，具体如下：

1. upper():将所有字符中的小写字母转换为大写字母
2. lower()：把所有字符中的大写字母转为小写字母
3. capitalize()：把第一个字母转化为大写字母，其余小写
4. title()：把每个单词的第一个字母转化为大写，其余小写

#### 【程序实现】

str = "www.runoob.com"  
 print(str.upper()) # 把所有字符中的小写字母转换成大写字母  
 print(str.lower()) # 把所有字符中的大写字母转换成小写字母  
 print(str.capitalize()) # 把第一个字母转化为大写字母，其余小写  
 print(str.title()) # 把每个单词的第一个字母转化为大写，其余小写

#### 【程序结果】

WWW.RUNOOB.COM

www.runoob.com

Www.runoob.com

Www.Runoob.Com

## 程序19： 获取昨天日期

#### 【程序描述】

通过导入 datetime 模块来获取昨天的日期。

#### 【程序分析】

#### 【程序实现】

# 引入 datetime 模块  
 import datetime  
 def getYesterday():  
 today=datetime.date.today()  
 oneday=datetime.timedelta(days=1)  
 yesterday=today-oneday  
 return yesterday  
 # 输出  
 print(getYesterday())

#### 【程序结果】

2017-03-02

## 程序20： 变量交换

#### 【程序描述】

编写一个程序，将用户输入的两个变量进行相互交换。要求不使用临时变量实现。

#### 【程序分析】

与其他语言不同，Python中的变量可以通过下列方式进行赋值：

x,y = y,x

上述代码的作用就是将x的值赋给y，将y的值赋给x，即x和y值的互换。

#### 【程序实现】

# 用户输入

x = input('输入 x 值: ')

y = input('输入 y 值: ')

# 不使用临时变量

x,y = y,x

print('交换后 x 的值为: {}'.format(x))

print('交换后 y 的值为: {}'.format(y))

#### 【程序结果】

输入 x 值: 3

输入 y 值: 4

交换后 x 的值为: 4

交换后 y 的值为: 3

## 程序21： 质数判断

#### 【程序描述】

一个大于1的自然数，除了1和它本身外，不能被其他自然数（质数）整除（2, 3, 5, 7等），换句话说就是该数除了1和它本身以外不再有其他的因数。

#### 【程序实现】

# 用户输入数字  
 num = int(input("请输入一个数字: "))  
 # 质数大于 1  
 if num > 1:  
 # 查看因子  
 for i in range(2, num):  
 if (num % i) == 0:  
 print(num, "不是质数")  
 print(i, "乘于", num // i, "是", num)  
 break  
 else:  
 print(num, "是质数")  
 # 如果输入的数字小于或等于 1，不是质数  
 else:  
 print(num, "不是质数")

#### 【程序结果】

请输入一个数字: 12

12 不是质数

2 乘于 6 是 12

## 程序22： 简单计算器

#### 【程序描述】

编写一个程序，实现简单计算器实现，包括两个数基本的加减乘除运算。

#### 【程序分析】

简单计算器的开发流程如下：

1. 打印菜单，提示用户选择要进行的运算
2. 用户选择运算类型
3. 接收用户输入的值，用于参与运算
4. 输出运算的结果

需要注意的是，由于接受的用户输入是字符串，需要将字符串转为数值类型

#### 【程序实现】

# 定义函数

def add(x, y):

"""相加"""

return x + y

def subtract(x, y):

"""相减"""

return x - y

def multiply(x, y):

"""相乘"""

return x \* y

def divide(x, y):

"""相除"""

return x / y

# 用户输入

print("选择运算：")

print("1、相加")

print("2、相减")

print("3、相乘")

print("4、相除")

choice = input("输入你的选择(1/2/3/4):")

num1 = int(input("输入第一个数字: "))

num2 = int(input("输入第二个数字: "))

if choice == '1':

print(num1,"+",num2,"=", add(num1,num2))

elif choice == '2':

print(num1,"-",num2,"=", subtract(num1,num2))

elif choice == '3':

print(num1,"\*",num2,"=", multiply(num1,num2))

elif choice == '4':

print(num1,"/",num2,"=", divide(num1,num2))

else:

print("非法输入")

#### 【程序结果】

加法运算：

选择运算：

1、相加

2、相减

3、相乘

4、相除

输入你的选择(1/2/3/4):1

输入第一个数字: 2

输入第二个数字: 3

2 + 3 = 5

减法运算：

选择运算：

1、相加

2、相减

3、相乘

4、相除

输入你的选择(1/2/3/4):2

输入第一个数字: 5

输入第二个数字: 1

5 - 1 = 4

乘法运算：

选择运算：

1、相加

2、相减

3、相乘

4、相除

输入你的选择(1/2/3/4):3

输入第一个数字: 2

输入第二个数字: 5

2 \* 5 = 10

除法运算：

选择运算：

1、相加

2、相减

3、相乘

4、相除

输入你的选择(1/2/3/4):4

输入第一个数字: 10

输入第二个数字: 2

10 / 2 = 5.0

## 程序23： 最大公约数

#### 【程序描述】

编写一个程序，计算两个数的最大公约数。

#### 【程序实现】

# 定义一个函数  
 def hcf(x, y):  
 """该函数返回两个数的最大公约数"""  
 # 获取最小值  
 if x > y:  
 smaller = y  
 else:  
 smaller = x  
 for i in range(1,smaller + 1):  
 if((x % i == 0) and (y % i == 0)):  
 hcf = i  
 return hcf  
 # 用户输入两个数字  
 num1 = int(input("输入第一个数字: "))  
 num2 = int(input("输入第二个数字: "))  
 print( num1,"和", num2,"的最大公约数为", hcf(num1, num2))

#### 【程序结果】

输入第一个数字: 12

输入第二个数字: 3

12 和 3 的最大公约数为 3

## 程序24： 最小公倍数

#### 【程序描述】

编写一个程序，计算两个数的最小公倍数。

#### 【程序实现】

# 定义函数

def lcm(x, y):

# 获取最大的数

if x > y:

greater = x

else:

greater = y

while(True):

if((greater % x == 0) and (greater % y == 0)):

lcm = greater

break

greater += 1

return lcm

# 获取用户输入

num1 = int(input("输入第一个数字: "))

num2 = int(input("输入第二个数字: "))

print( num1,"和", num2,"的最小公倍数为", lcm(num1, num2))

#### 【程序结果】

输入第一个数字: 12

输入第二个数字: 2

12 和 2 的最小公倍数为 12

## 程序25： 字符串判断

#### 【程序描述】

编写一个程序，演示Python字符串中和判断有关的方法的使用。

#### 【程序分析】

字符串中常见的判断操作的方法如下所示：

1. isalnum(): 判断所有字符都是数字或者字母
2. isalpha(): 判断所有字符都是字母
3. isdigit():判断所有字符都是数字
4. islower():判断所有字符都是小写
5. isupper():判断所有字符都是大写
6. istitle():判断所有单词都是首字母大写，像标题
7. isspace():判断所有字符都是空白字符、\t、\n、\r

#### 【程序实现】

print("测试实例一")  
 str = "www.itheima.com"  
 print(str.isalnum()) # 判断所有字符都是数字或者字母  
 print(str.isalpha()) # 判断所有字符都是字母  
 print(str.isdigit()) # 判断所有字符都是数字  
 print(str.islower()) # 判断所有字符都是小写  
 print(str.isupper()) # 判断所有字符都是大写  
 print(str.istitle()) # 判断所有单词都是首字母大写，像标题  
 print(str.isspace()) # 判断所有字符都是空白字符、\t、\n、\r  
 print("------------------------")  
 # 测试实例二  
 print("测试实例二")  
 str = "itheima"  
 print(str.isalnum())  
 print(str.isalpha())  
 print(str.isdigit())  
 print(str.islower())  
 print(str.isupper())  
 print(str.istitle())  
 print(str.isspace())

#### 【程序结果】

测试实例一

False

False

False

True

False

False

False

------------------------

测试实例二

True

True

False

True

False

False

False

## 程序26： 合并文件数据

#### 【程序描述】

有两个磁盘文件A和B,各存放一行字母,要求把这两个文件中的信息合并(按字母顺序排列), 输出到一个新文件C中。

#### 【程序实现】

fp = open('test1.txt')  
 a = fp.read()  
 fp.close()

fp = open('test2.txt')

b = fp.read()

fp.close()

fp = open('test3.txt', 'w')

l = list(a + b)

l.sort()

s = ''

s = s.join(l)

fp.write(s)

fp.close()

#### 【程序结果】

假设test1.txt文件中存放的是hello，test2.txt文件中存放的是world，那么程序运行后，test3.txt文件中的数据是：dehllloorw

## 程序27： 猜数游戏

#### 【程序描述】

编写一个猜数游戏，该游戏会随机产生一个数字，用户可以随意输入一个数进行比较，在比较过程中，会不断提示用户输入的数是大了还是小了，直到用户输入的数等于随机数，程序终止。

#### 【程序实现】

import time  
import random  
play\_it = input('do you want to play it.(\'y\' or \'n\')')  
while play\_it == 'y':  
 c = input('input a character:\n')  
 i = random.randint(0, 2 \*\* 32) % 100  
 print('please input number you guess:\n')  
 start = time.clock()  
 a = time.time()  
 guess = int(input('input your guess:\n'))  
 while guess != i:  
 if guess > i:  
 print('please input a little smaller')  
 guess = int(input('input your guess:\n'))  
 else:  
 print('please input a little bigger')  
 guess = int(input('input your guess:\n'))  
 end = time.clock()  
 b = time.time()  
 var = (end - start) / 18.2  
 print(var)  
 if var < 15:  
 print('you are very clever!')  
 elif var < 25:  
 print('you are normal!')  
 else:  
 print('you are stupid!')  
 print('Congradulations')  
 print('The number you guess is %d' % i)  
 play\_it = input('do you want to play it.')

#### 【程序结果】

do you want to play it.('y' or 'n')y

input a character:

5

please input number you guess:

input your guess:

60

……

please input a little bigger

input your guess:

29

please input a little smaller

input your guess:

28

3.81868131868e-05

you are very clever!

Congradulations

The number you guess is 28

do you want to play it.

## 程序28：为数据加密

#### 【程序描述】

某个公司采用公用电话传递数据，数据是四位的整数，在传递过程中是加密的，加密规则如下：每位数字都加上5,然后用和除以10的余数代替该数字，再将第一位和第四位交换，第二位和第三位交换。

#### 【程序实现】

from sys import stdout  
a = int(input('input a number:\n'))  
aa = []  
aa.append(a % 10)  
aa.append(a % 100 / 10)  
aa.append(a % 1000 / 100)  
aa.append(a / 1000)  
for i in range(4):  
 aa[i] += 5  
 aa[i] %= 10  
for i in range(2):  
 aa[i],aa[3 - i] = aa[3 - i],aa[i]  
for i in range(3,-1,-1):  
 stdout.write(str(aa[i]))

#### 【程序结果】

input a number:

12

76.25.125.012

## 程序29：平方运算

#### 【程序描述】

编写程序，求输入数字的平方，如果平方运算后小于 50 则退出。

#### 【程序实现】

TRUE = 1  
FALSE = 0  
def SQ(x):  
 return x \* x  
print('如果输入的数字小于 50，程序将停止运行。')  
again = 1  
while again:  
 num = int(input('Please input number'))  
 print('运算结果为 %d' % (SQ(num)))  
 num=SQ(num)  
 if num >= 50:  
 again = TRUE  
 else:  
 again = FALSE

#### 【程序结果】

如果输入的数字小于 50，程序将停止运行。

Please input number12

运算结果为 144

Please input number12

运算结果为 144

Please input number3

运算结果为 9

## 程序30： 计算0-7组成的奇数个数

#### 【程序描述】

编写一个程序，计算0—7所能组成的奇数个数。

#### 【程序实现】

sum = 4  
s = 4  
for j in range(2,9):  
 print(sum)  
 if j <= 2:  
 s \*= 7  
 else:  
 s \*= 8  
 sum += s  
print('sum = %d' % sum)

#### 【程序结果】

4

32

256

2048

16384

131072

1048576

sum = 8388608

## 程序31：求值

#### 【程序描述】

809\*??=800\*??+9\*??+1 其中??代表的两位数,8\*??的结果为两位数，9\*??的结果为3位数。求??代表的两位数，及809\*??后的结果。

#### 【程序实现】

a = 809  
for i in range(10,100):  
 b = i \* a + 1  
 if b >= 1000 and b <= 10000 and 8 \* i < 100 and 9 \* i >= 100:  
 print(b,'/',i,' = 809 \* ',i,' + ', b % i)

#### 【程序结果】

9709 / 12 = 809 \* 12 + 1

## 程序32：猴子分桃

#### 【程序描述】

海滩上有一堆桃子，五只猴子来分。第一只猴子把这堆桃子平均分为五份，多了一个，这只猴子把多的一个扔入海中，拿走了一份。第二只猴子把剩下的桃子又平均分成五份，又多了一个，它同样把多的一个扔入海中，拿走了一份，第三、第四、第五只猴子都是这样做的，问海滩上原来最少有多少个桃子？

#### 【程序实现】

i = 0  
j = 1  
x = 0  
while (i < 5):  
 x = 4 \* j  
 for i in range(0, 5):  
 if (x % 4 != 0):  
 break  
 else:  
 i += 1  
 x = (x / 4) \* 5 + 1  
 j += 1  
print(x)

#### 【程序结果】

3121.0

## 程序33：淘汰游戏

#### 【程序描述】

有n个人围成一圈，顺序排号。从第一个人开始报数（从1到3报数），凡报到3的人退出圈子，问最后留下的是原来第几号的那位。

#### 【程序实现】

nmax = 50  
n = int(input('请输入总人数:'))  
num = []  
for i in range(n):  
 num.append(i + 1)  
i = 0  
k = 0  
m = 0  
while m < n - 1:  
 if num[i] != 0 : k += 1  
 if k == 3:  
 num[i] = 0  
 k = 0  
 m += 1  
 i += 1  
 if i == n : i = 0  
i = 0  
while num[i] == 0: i += 1  
print(num[i])

#### 【程序结果】

请输入总人数:23

8

## 程序34：分解质因子

#### 【程序描述】

编写一个程序，用于对一个数进行分解质因子。

#### 【程序实现】

n=int(input(**'please input n:'**))   
result=[]   
i=2   
str1=str(n)+'='   
while n>1:   
 if n%i==0:   
 n/=i   
 result.append(str(i))   
 i -= 1   
 i +=1   
str1+='\*'.join(result)   
print(str1)

#### 【程序结果】

please input n:16

16=2\*2\*2\*2

## 程序35：取整数右端的4~7位

#### 【程序描述】

编写程序，取一个整数a从右端开始的4〜7位。

#### 【程序分析】

可以这样考虑：

(1)先使a右移4位。

(2)设置一个低4位全为1,其余全为0的数。可用~(~0<<4)

(3)将上面二者进行&运算。

#### 【程序实现】

a = int(input('input a number:\n'))  
b = a >> 4  
c = ~(~0 << 4)  
d = b & c  
print('%o\t%o' % (a, d))

#### 【程序结果】

input a number:

9

11 0

## 程序36：判断4位回文数

#### 【程序描述】

用户输入一个4位的整数，如果是回文数显示True，如果不是回文数显示False。

#### 【程序分析】

回文数，就是反过来的数字和正着的数字是相同的，如1221，倒过来的数还是1221，这就是一个回文数，1234倒过来是4321，不相等，就不是回文数。

一个四位数abcd可以写为I1=a\*1000+b\*100+c\*10+d

那么倒过来的数就可以写成I2=d\*1000+c\*100+b\*10+a

如果I1和I2相等，那么这就是一个回文数。I1==I2的结果就是True，否则就是False。

一个四位数I1，分离各位数字的方法是：

千位：I1/1000

百位：I1/100%10

十位：I1/10%10

个位：I1%10

因此，判断回文数，如果使用a、b、c、d表示整数I1的千位、百位、十位、个位，开发步骤如下：

1. 用户输入一个整数I1
2. 分离千位、百位、十位、个位
3. 组合出新数字I2=d\*1000+c\*100+b\*10+a
4. 比较I1==I2
5. 输出结果

#### 【程序实现】

I1=int(input("请输入一个四位数："))  
a=int(I1/1000)  
b=int(I1/100%10)  
c=int(I1/10%10)  
d=int(I1%10)  
I2=d\*1000+c\*100+b\*10+a  
if I1==I2:  
 print("True")  
else:  
 print("False")

#### 【程序结果】

例如，输入1221，结果输出True

请输入一个四位数：1221

True

输入1234，结果输出False

请输入一个四位数：1234

False

## 程序37：汉诺塔（递归函数）

#### 【程序描述】

古代有一个梵塔，塔内有A、B、C三个基座，A座上有64个盘子，盘子大小不等，大的在下，小的在上。有人想把这64个盘子从A座移到C座，但每次只允许移动一个盘子，并且在移动的过程中，3个基座上的盘子始终保持大盘在下，小盘在上。在移动过程中盘子可以放在任何一个基座上，不允许放在别处。编写程序，用户输入盘子的个数，显示移动的过程。

#### 【程序分析】

假定盘子从大到小依次编号为：盘1、盘2、…

1. 如果只有一个盘子，则不需要利用B座，直接将盘子从A移动到C
2. 如果有2个盘子，可以先将盘2移动到B，将盘1移动到C后，再将盘2移动到C
3. 如果有3 个盘子，那么根据2个盘子的结论，可以借助C将盘2和盘3从A移动到B，将盘1从A移动到C，A变成空座；借助A座，将B上的两个盘子移动到C

上述思路可以一直扩展下去，根据以上的分析，可以写出下面的递归表达：

将一个盘子从A移动到C

借助C将n-1个盘子从A移动到B

将一个盘子从A移动到C n>1

借助A将n-1个盘子从B移动到C

借助B将n个盘子从A移动到C

为了编写一个递归函数实现“借助B将n个盘子从A移到C”，比较等式左右两边相似操作，会发现：

1. 盘子的数量从n变化到n-1，问题规模缩小了，显然n是一个可变的参数
2. 盘子的起始位置是变化的，等式左侧是A，右侧是A或B
3. 盘子的最终位置是变化的，等式左侧是C、右侧是B或C
4. 同样被借助的位置也是变化的

因此，递归函数共有盘子数、起始位置、借助位置和最终位置4个变量，因此函数有4个可变参数。假定函数的参数一次为盘子数、初始位置、借助位置和最终位置，则可写出下面函数。

#### 【程序实现】

def Hanoi(n,ch1,ch2,ch3):  
 if n==1:  
 print(ch1,'->',ch3)  
 else:  
 Hanoi(n-1,ch1,ch3,ch2)  
 print(ch1,'->',ch3)  
 Hanoi(n-1,ch2,ch1,ch3)  
N=int(input("请输入盘子的数量："))  
Hanoi(N,'A','B','C')

#### 【程序结果】

请输入盘子的数量：4

A -> B

A -> C

B -> C

A -> B

C -> A

C -> B

A -> B

A -> C

B -> C

B -> A

C -> A

B -> C

A -> B

A -> C

B -> C

## 程序38：判断闰年

#### 【程序描述】

用户输入一个年份，如果是闰年输出True，如果不是，输出False。

#### 【程序分析】

判断闰年的规则如下:

1. 能被4整除且不能被100整除的为闰年
2. 能被400整除的是闰年

在上述两个规则中，只要有一条规则成立，则年份为闰年。

#### 【程序实现】

#### 【程序结果】

## 程序39：寻找自幂数（38）

#### 【程序描述】

用户输入位数n，找出并显示出所有n位的自幂数。

#### 【程序分析】

自幂数是指一个n位正整数，如果它的各位数字的n次方的和加起来等于这个数，数学家称这样的数为自幂数。例如，13+53+33=153，153就是一个3位的自幂数，3位自幂数也称为水仙花数。本程序设n的数值为1~6，当n大于6时，程序退出。

#### 【程序实现】

#自幂数  
start=0  
end=0  
digit=0  
m=0  
n=int(input('请选择自幂数的位数【1，2，3，4，5，6】:'))  
while 0<n<7:  
 start=pow(10,n-1)  
 end=pow(10,n)-1  
 print(n,'位数的自幂数有:')  
 for k in range(start,end+1):  
 m=k  
 total=0  
 while m!=0:  
 digit=m%10  
 total+=pow(digit,n)  
 m=m//10  
 if total==k:  
 print(str(k),end=' ')  
 n=int(input('\n 请选择自幂数的位数【1，2，3，4，5，6】:'))  
else:  
 print('输入位数不在范围内,程序结束。')

#### 【程序结果】

请选择自幂数的位数【1，2，3，4，5，6】:1

1 位数的自幂数有:

1 2 3 4 5 6 7 8 9

请选择自幂数的位数【1，2，3，4，5，6】:2

2 位数的自幂数有:

请选择自幂数的位数【1，2，3，4，5，6】:3

3 位数的自幂数有:

153 370 371 407

请选择自幂数的位数【1，2，3，4，5，6】:4

4 位数的自幂数有:

1634 8208 9474

请选择自幂数的位数【1，2，3，4，5，6】:5

5 位数的自幂数有:

54748 92727 93084

请选择自幂数的位数【1，2，3，4，5，6】:6

6 位数的自幂数有:

548834

请选择自幂数的位数【1，2，3，4，5，6】:

## 程序40：多维数据

#### 【程序描述】

在现实生活中，经常需要处理多维数据，比如，二维矩阵，三维矩阵等。在Python中可以用嵌套列表来表示多维数据，比如，下面的一个列表表示的是3\*3的二维数据。

M=[[1,2,3],[‘a’,’b’,’c’],[7,8,9]]

显然这种嵌套列表与通常意义的矩阵有所不同，主要不同表现在以下两个方面。

1. 列表的每个维度长度可以不同
2. 列表的元素数据类型可以不同

尽管如此，使用嵌套列表表示多维矩阵依然是Python语句中的常见用法，比如列表N=[[1,2,3],[4,5,6],[7,8,9]]表示下面矩阵。

123

456

789

而对于第（i,j）位置的访问使用N[i][j]即可。

如果要通过键盘输入数据来初始化一个2\*2的矩阵，可以使用下面的语句。

#### 【程序分析】

#### 【程序实现】

a=[]  
for i in range(2):  
 a.append([])  
 for j in range(2):  
 v=int(input("请输入元素："))  
 a[i].append(v)  
print(a)

#### 【程序结果】

请输入元素：1

请输入元素：2

请输入元素：2

请输入元素：3

[[1, 2], [2, 3]]

## 程序41： 验证哥德巴赫猜想

#### 【程序描述】

哥德巴赫猜想说是说，任何一个超过2的偶数都可以写成两个素数之和，例如，4=2+2，8=5+3等。本例要求根据用户输入的偶数找出其素数和的分解形式

#### 【程序分析】

一个简单的方法的，对于输入的偶数N，找出其所有分解，逐一验证每一个满足N=k1+k2的分解中k1和k2是否都是素数。比如对于数字12，验证分解（2，10），（3，9）、（4，8）、（5、7）、（6，6）中有没有两个数都是素数的情形。如果有，哥德巴赫猜想该数就是成立的。这种算法对于只验证一个数字N的所有分解的情形是合适的。但对于需要验证多个偶数N的情形效率欠佳。比如需要验证10、12、16三个数，它们有分解5+5、5+7、5+11，这样验证这几个分解时就要判断5是不是素数，重复的运算会很多。

本案例采用另一种思路，首先建立一个素数表，该素数表要足够长，可以覆盖偶数N所有分解中可能遇到的素数。而后考察N的每个分解，看看分解出来的两个数是否都包含在素数表中，若是，则找到一种素数分解。

#### 【程序实现】

def main():  
 # 输入待验证的偶数  
 N=int(input("请输入待验证的偶数："))  
 while N<3 or N%2==1:  
 print("输入的数不符合要求")  
 N=int(input("请输入待验证的偶数n（n>2）："))  
 #生成素数表  
 Prime=set()  
 for i in range(2,N+1):  
 Prime.add(i)  
 for i in range(2,N+1):  
 if i in Prime:  
 for k in range(2\*i,N+1,i):  
 if k in Prime:  
 Prime.remove(k)  
 #验证该偶数能否分解为两个素数之和  
 for e in Prime:  
 f=N-e  
 if f>=e and f in Prime:  
 print(N,'=',e,'+',f)  
main()

#### 【程序结果】

请输入待验证的偶数：12

12 = 5 + 7

## 程序42： 解方程

#### 【程序描述】

编写程序，解一元二次方程 ax2+bx+c=0。

#### 【程序分析】

#### 【程序实现】

from math import \*  
print("本程序求 ax^2+bx+c=0 的根")  
a=float( input("请输入 a:") )  
b=float( input("请输入 b:") )  
c=float( input("请输入 c:") )  
delta=b\*b-4\*a\*c  
if(delta>=0):  
 delta=sqrt(delta)  
 x1=(-b+delta)/2/a  
 x2=(-b-delta)/(2\*a)  
 print("两个实根分别为:",x1,x2)  
else:  
 print("没有实根")

#### 【程序结果】

本程序求 ax^2+bx+c=0 的根

请输入 a:12

请输入 b:23

请输入 c:21

没有实根

或

本程序求 ax^2+bx+c=0 的根

请输入 a:2

请输入 b:-8

请输入 c:2

两个实根分别为: 3.732050807568877 0.2679491924311228

## 程序43： 异常处理

#### 【程序描述】

输入两个整数，打印它们相除之后的结果。对输入的不是整数或除数为零，进行异常处理。

#### 【程序分析】

在对两个整数相除过程中，如果遇到输入的不是整数或者除数为零，那么程序会终止，不再执行其他语句，这就是出现了异常。系统自动的处理就是停止执行，给出提示，为了执行这种情况，不让程序结束，可以将会出现异常的代码写在try中，让except按照异常类名处理异常。

#### 【程序实现】

k=0  
while(k<3):  
 try:  
 x=int(input('请输入第一个整数:'))  
 y=int(input('请输入第二个整数:'))  
 print('x/y=',x/y)  
 except ValueError:  
 print('请输入一个整数。')  
 except ZeroDivisionError:  
 print('除数不能为零。')  
 k=k+1

#### 【程序结果】

请输入第一个整数:12

请输入第二个整数:23

x/y= 0.5217391304347826

## 程序44： 统计不及格人数（continue语句）

#### 【程序描述】

有若干成绩，统计及格人的平均成绩。

#### 【程序分析】

有若干成绩，其中有不及格的，现在要统计及格的这些人的平均成绩，可以使用循环先求和，但遇到不及格的，不加入其中，最后除以及格的人数，假设成绩存储在一个列表中。

#### 【程序实现】

x=[98,72,80,45,30,89,92,54,48,82,67,76]  
sum=0  
k=0;  
for item in x:  
 if(item<60):  
 continue  
 sum=sum+item  
 k=k+1;  
if(k!=0):  
 print("及格人数",k,"人,平均成绩是",sum/k)

#### 【程序结果】

及格人数 8 人,平均成绩是 82.0

## 程序45： 求直角三角形的斜边

#### 【程序描述】

输入直角三角形的两个直角边长度a、b，求斜边c的长度。

数学公式 c=

#### 【程序分析】

在Python中，乘法使用“\*”号表示，如a\*a就是a的平方。

除此之外，还有开方根的运算，可以通过Python中math模块内置的函数来实现，为此，我们可以在程序中使用import导入math模块，然后调用开方的函数sqrt实现。

#### 【程序实现】

###################################  
# 勾股定理求斜边长度  
# python0102.py  
# 2015  
###################################  
from math import \*  
a=float(input("请输入斜边 1 的长度")) #输入实数  
b=float(input("请输入斜边 2 的长度")) #输入实数  
c=a\*a+b\*b #计算,得到的是斜边的平方  
c=sqrt(c) #开方，得到的是斜边长  
print("斜边长为:",c) #显示，一项是字符串，一项是 c 表示的斜边长

#### 【程序结果】

请输入斜边 1 的长度3

请输入斜边 2 的长度4

斜边长为: 5.0

## 程序46： 问年龄

#### 【程序描述】

有5个人坐在一起，问第五个人多少岁？他说比第4个人大2岁。问第4个人岁数，他说比第3个人大2岁。问第三个人，又说比第2人大两岁。问第2个人，说比第一个人大两岁。最后问第一个人，他说是10岁。请问第五个人多大？

#### 【程序分析】

题目非常符合递归函数的过程，所以本题采用递归函数完成。

函数把问到的第几个人作为参数i，当i的值为1时，函数返回10，其它情况均为第i-1人年龄的基础上加2.

#### 【程序实现】

def fun(i):

if i==1:

return 10

return fun(i-1)+2

print(fun(5))

#### 【程序结果】

18

## 程序47：求相同数字的和

#### 【程序描述】

求s=a+aa+aaa+aaaa+aa...a的值，其中a是一个数字。例如2+22+222+2222+22222(此时共有5个数相加)，几个数相加由键盘控制。

#### 【程序分析】

根据用户输入的个数（长度）决定循环的次数。

区分每次循环输出的内容，只要是最后一次循环，就直接输出最终的结果，其余皆为本次结果跟上加号。

先从个位的数字开始计算，每执行一次循环就把上次的结果数乘以10，使得每次的结果为30、330、3330，然后再加上个位的基本数字就行。

#### 【程序实现】

def test():  
 basis = int(input("输入一个基本的数字:"))  
 n = int(input("输入数字的长度:"))  
 b = basis  
 sum = 0  
 for i in range(0, n):  
 if i == n - 1:  
 print("%d " % (basis))  
 else:  
 print("%d +" % (basis))  
 sum += basis  
 basis = basis \* 10 + b  
 print('= %d' % (sum))  
test()

#### 【程序结果】

输入一个基本的数字:3

输入数字的长度:4

3 +

33 +

333 +

3333

= 3702

## 程序48：找出1000以内的所有完数

#### 【程序描述】

一个数如果恰好等于它的因子之和，这个数就称为“完数”。例如6=1＋2＋3.编程找出1000以内的所有完数。

提示：借助math模块的sqrt函数（求平方根）

#### 【程序分析】

#### 【程序实现】

from math import sqrt  
n = int(input('输入一个数字:'))  
sum = n\*-1  
k = int(sqrt(n))  
for i in range(1,k+1):  
 if n%i == 0:  
 sum += n/i  
 sum += i  
if sum == n:  
 print('YES')  
else:  
 print('NO')

#### 【程序结果】

输入一个数字:6

YES

## 程序49：公民类

#### 【程序描述】

定义公民类，实例成员变量有身份证号、姓名、年龄和性别。定义公民类的派生类:学生类和教师类。学生类增加实例成员变量学号、班级和分数;教师类增加实例变量工号、系别和工资。编写主程序， 定义类的对象，设置对象的实例属性，显示对象的信息。

#### 【程序分析】

使用super()方法调用父类的构造方法。

#### 【程序实现】

class Citizen:  
 def \_\_init\_\_(self,id,name,age,sex):  
 self.id = id  
 self.name = name  
 self.age = age  
 self.sex = sex  
class Student(Citizen):  
 def \_\_init\_\_(self,id,name,age,sex,stdno,grade,score):  
 super(Student, self).\_\_init\_\_(id,name,age,sex)  
 self.stdno = stdno  
 self.grade = grade  
 self.score = score  
class Teacher(Citizen):  
 def \_\_init\_\_(self,id,name,age,sex,thno,dept,salary):  
 super(Teacher, self).\_\_init\_\_(id,name,age,sex)  
 self.thno = thno  
 self.dept = dept  
 self.salary = salary #主程序  
c=Citizen('101','zhang',20,'female')  
print(c.id,c.name,c.age,c.sex)  
s=Student('102','wang',30,'male',1221,'computer12',630)  
print(s.id,s.name,s.age,s.sex,end=" ")  
print(s.stdno,s.grade,s.score)  
t=Teacher('103','li',36,'male',356,'eie',4500)  
print(t.id,t.name,t.age,t.sex,end=" ")  
print(t.thno,t.dept,t.salary)

#### 【程序结果】

101 zhang 20 female

102 wang 30 male 1221 computer12 630

103 li 36 male 356 eie 4500

## 程序50：复数类

#### 【程序描述】

定义一个复数类，包括实部和虚部成员变量、构造方法、以及两个复数的加法、乘法和比 较大小运算符定义。

#### 【程序实现】

class Complex:  
 def \_\_init\_\_(self,r,i):  
 self.real = r  
 self.imag = i  
 def \_\_add\_\_(self, c):  
 return Complex(self.real + c.real, self.imag + c.imag)  
 def \_\_mul\_\_(self, c):  
 return Complex(self.real \* c.real- self.imag \* c.imag,

self.real \* c.imag+ self.imag \* c.real)  
 def \_\_gt\_\_(self, c):  
 if self.real>c.real:  
 return True  
 elif self.real<c.real:  
 return False  
 elif self.imag>c.imag:  
 return True  
 else:  
 return False  
 def show(self):  
 print(self.real,"+",self.imag, "j")  
#主函数  
c1 = Complex(3,4)  
c2 = Complex(6,-7)  
(c1 + c2).show()  
(c1 \* c2).show()  
print(c1 > c2)

#### 【程序结果】

9 + -3 j

46 + 3 j

False