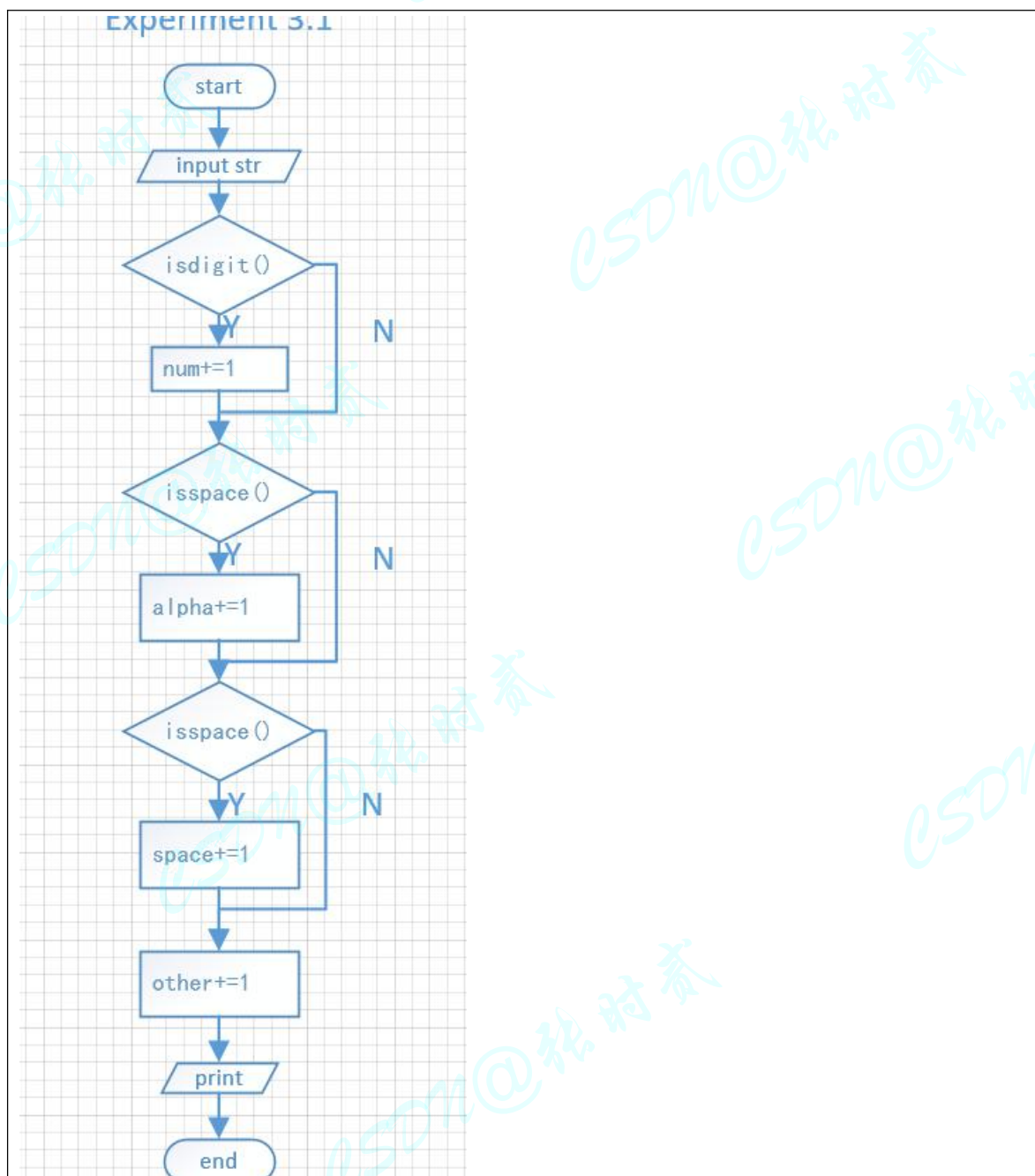


实验环境	严禁抄袭 仅供参考 Blog:zhangshier.vip
<b>一、实验题目</b> 实验三：函数	
<b>二、实验目的</b> <ol style="list-style-type: none"> <li>(1) 掌握函数的定义和调用方法。</li> <li>(2) 掌握函数参数传递原理和方法。</li> <li>(3) 掌握匿名函数、嵌套函数、递归函数的创建和调用方法。</li> <li>(4) 掌握变量类型和作用域。</li> </ol>	
<b>三、实验内容</b> <ol style="list-style-type: none"> <li>1、编写一个函数 <code>func(str)</code>，计算并返回字符串 <code>str</code> 中的数字、字母及其他类型字符的个数。</li> <li>2、验证哥德巴赫猜想：任意一个大于 2 的偶数都可以表示成 2 个素数之和。编写一个函数 <code>isGDBH(n)</code> 将传入的 6~100 之间的偶数表示为 2 个素数之和，结果保存在列表中返回。例如，函数传入参数 10，则返回["10=3+7", "10=5+5"]。</li> <li>3、编写实现如下功能：               <ol style="list-style-type: none"> <li>(1) 编写 3 个函数，分别求三角形、矩形和圆形周长。</li> <li>(2) 使用装饰器对上述 3 个函数的传入参数进行调用和合法性检查。</li> </ol> </li> </ol>	
<b>四、实验步骤</b> <ol style="list-style-type: none"> <li>1、题目一               <ol style="list-style-type: none"> <li>(1) 问题分析（含解决思路、使用的数据结构、程序流程图等）                   <p>在 <code>func</code> 函数中利用元组定义数字、字母、空格、其它，并初始化为 0。之后利用 <code>for</code> 循环对字符遍历，依次 <code>if</code> 判断是哪种字符类型</p> </li> </ol> </li> </ol>	



(2) 算法/代码描述（基本要求源代码）

content = input（'输入字符串'）

def func(s):

num = 0

alpha = 0

space = 0

others = 0

dic = {

    'num': 0,

    'alpha': 0,

    'space': 0,

    'other': 0

```

    }
    for i in s:
        if i.isdigit():
            dic['num'] += 1
        elif i.isalpha():
            dic['alpha'] += 1
        elif i.isspace():
            dic['space'] += 1
        else:
            dic['other'] += 1
    return dic
print(func(content))

```

(3) 运行结果（含执行结果验证、输出显示信息）

```

"D:\太工\大三第二学期\Python\192062116 张帆\venv\Script
输入字符串abc123  +++
{'num': 3, 'alpha': 3, 'space': 3, 'other': 3}

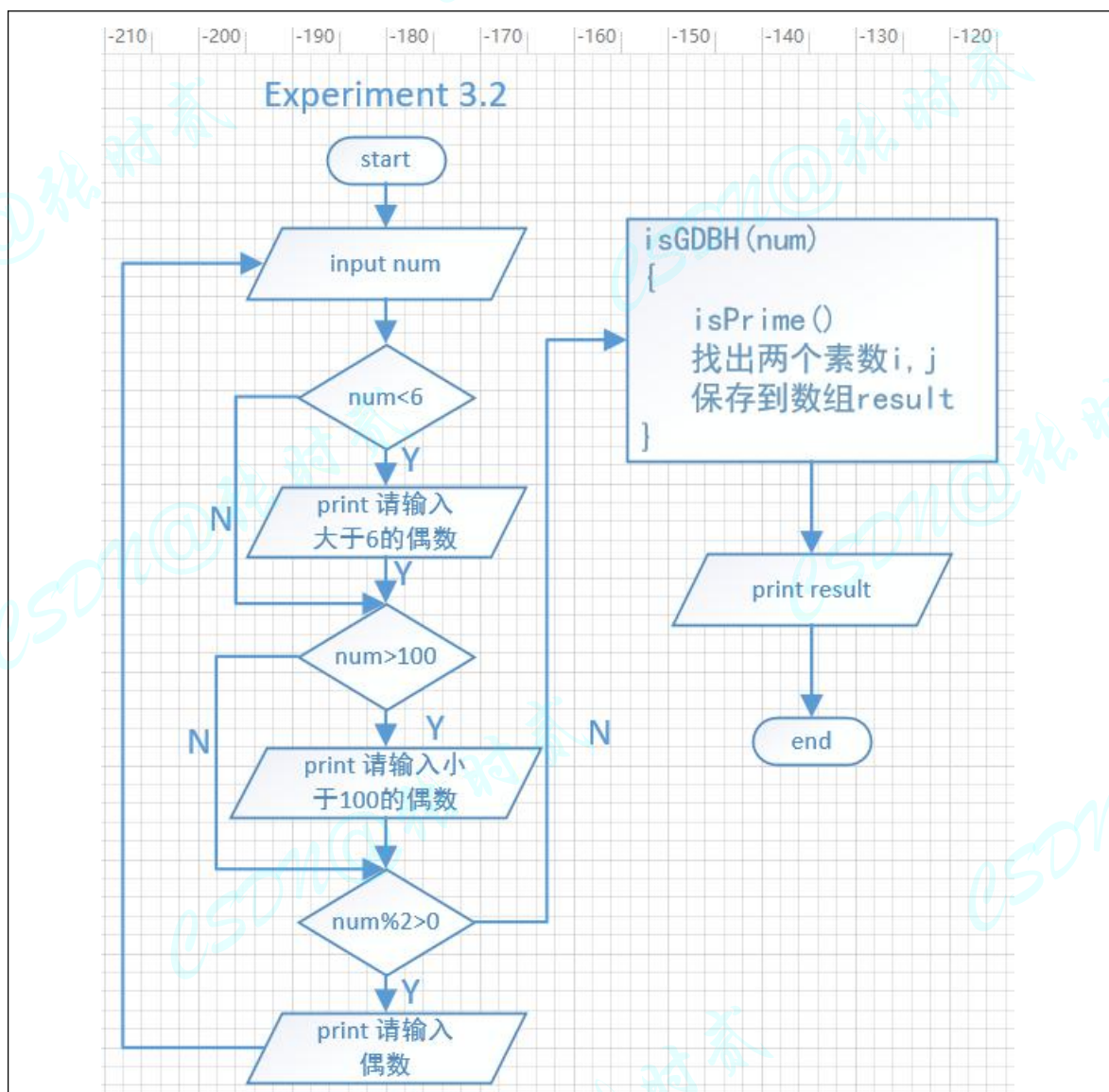
进程已结束，退出代码 0

```

## 2、题目二

(1) 问题分析（含解决思路、使用的数据结构、程序流程图等）

输入一个数，验证输入的数是否在 6~100 之间，调用 isGDBH(n)函数验证数据并将结果保存到列表，其中利用 isPrime(n)先判断一下是否为素数



(2) 算法/代码描述（基本要求源代码）

```

#判断是否为素数
def isPrime(n):
    i=2
    while i<=n:
        if n%i==0:
            break
        i+=1
    if n==i:
        return True
num=int(input("输入偶数，6~100 之间\n"))
if num<6:
    print("请输入大于 6 的偶数！")
    exit(0)
elif num>100:

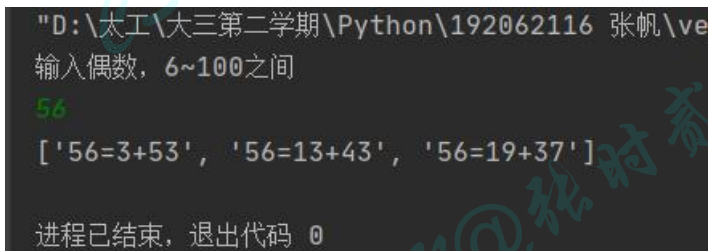
```

```

print("请输入小于 100 的偶数！")
exit(0)
elif (num%2)>0:
    print("请输入偶数！")
    exit(0)
i=1
result = []
def isGDBH(n):
    global i
    while(i<=num):
        i = i + 1
        if(isPrime(i)):
            j=1
            while(j<num):
                j = j + 1
                if(isPrime(j) and i<=j): #j<i 防止重复
                    if(j+i==num):
                        elem = str(num)+'='+str(i)+'+'+str(j)
                        result.append(elem)
isGDBH(num)
print(result)

```

### (3) 运行结果（含执行结果验证、输出显示信息）



```

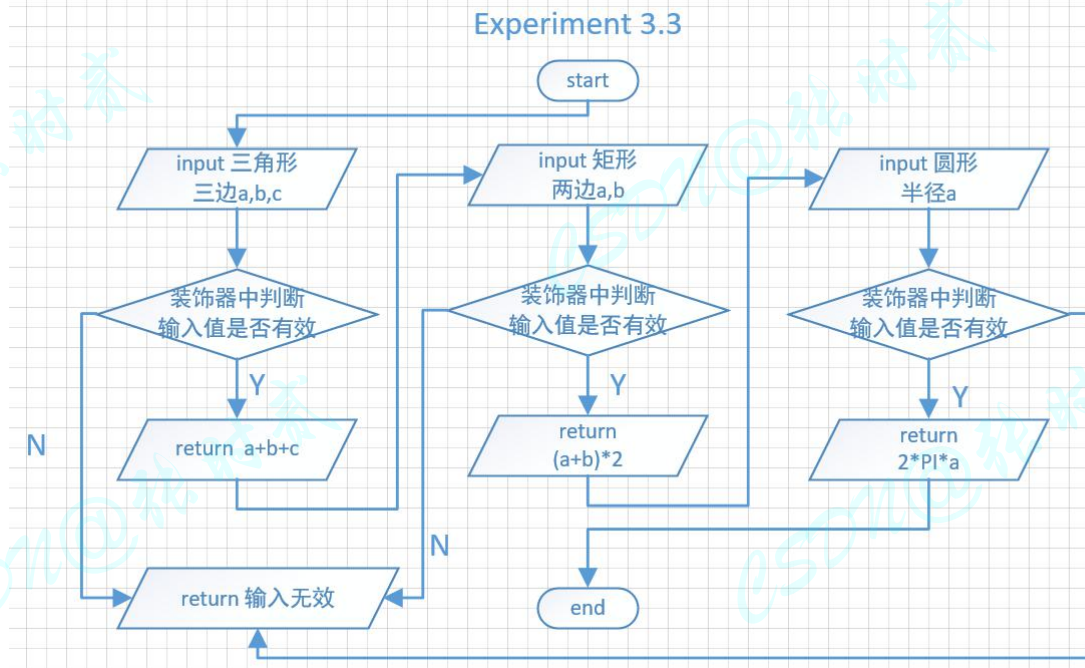
"D:\太工\大三第二学期\Python\192062116 张帆\ve
输入偶数，6~100之间
56
['56=3+53', '56=13+43', '56=19+37']
进程已结束，退出代码 0

```

### 3、题目三

#### (1) 问题分析（含解决思路、使用的数据结构、程序流程图等）

设计三个函数去接收输入的三种值，通过带参的装饰器去分别调用各自的判断条件，通过判断 **return**



(2) 算法/代码描述（基本要求源代码）

```

import math
#定义带参数的装饰器
def DECO(Shape):
    #定义内部装饰器
    def deco(func):
        #定义内函数
        def call_func(a,b=0,c=0):
            if(Shape=='三角形'):
                # print('测试')
                if ((a + b > c) & (a + c > b) & (b + c > a)):
                    return func ( a, b, c )
                else:
                    return '输入无效'
            if(Shape=='矩形'):
                if((a>0)&(b>0)):
                    return func(a,b)
                else:
                    return '输入无效'
            if(Shape=='圆形'):
                if(a>0):
                    return func(a)
                else:
                    return '输入无效'
            return call_func
        return call_func
  
```



```

    return deco
#传递装饰器参数
@DECO('三角形')
def Triangle(a,b,c):
    return a+b+c

@DECO('矩形')
def Cube(a,b):
    return (a+b)*2

@DECO('圆形')
def Circle(a):
    return 2*math.pi*a

if __name__ == "__main__":
    a,b,c=map(int,input('输入三角形三边').split(" "))
    print(Triangle(a,b,c))
    a,b=map(int,input('输入矩形两边').split(" "))
    print(Cube(a,b))
    a=int(input('输入圆半径'))
    print(Circle(a))

```

### (3) 运行结果（含执行结果验证、输出显示信息）

输入有效

```

"D:\太工\大三第二学期\Python\192062116 张帆\venv\
输入三角形三边3 4 5
12
输入矩形两边2 2
8
输入圆半径1
6.283185307179586

```

输入无效判断

```

"D:\太工\大三第二学期\Python\192062116 张帆\
输入三角形三边-1 1 1
输入无效
输入矩形两边-1 2
输入无效
输入圆半径-1
输入无效

```

## 五、 出现的问题及解决的方法

```
Traceback (most recent call last):  
  File "D:\太工\大三第二学期\Python\Experiment\Experiment 3.2.py", line 39, in <module>  
    isGDBH(a)  
  File "D:\太工\大三第二学期\Python\Experiment\Experiment 3.2.py", line 28, in isGDBH  
    while(i<=a):  
UnboundLocalError: local variable 'i' referenced before assignment
```

i 是全局变量，在函数中需要声明 `global i`