

第一题

请改正下面程序中存在的错误。

```
def StudentInfo(country='China', chineselevel='A', name):  
    print('%s,%s,%s'%(name,country,chineselevel))  
StudentInfo(country='America',  
chineselevel='B', name='John')
```

改正:

```
def StudentInfo(country='China', chineselevel='A', name='A'):  
    print('%s,%s,%s'%(name,country,chineselevel))  
StudentInfo(country='America', chineselevel='B', name='John')
```

1 请改正下面程序中存在的错误。

```
def StudentInfo(country='China', chineselevel='A', name):  
    print('%s,%s,%s'%(name,country,chineselevel))  
StudentInfo(country='America', chineselevel='B', name='John') (2分)
```

源代码内容:

请改正下面程序中存在错误。

```
def StudentInfo(country='China', chineselevel='A', name):  
    print('%s,%s,%s'%(name,country,chineselevel))  
StudentInfo(country='America', chineselevel='B', name='John')
```

输出样例:

John,America,B

时间限制: 500ms 内存限制: 32000kb

Python3.4

```
def StudentInfo(country='China', chineselevel='A', name='A'):  
    print('%s,%s,%s'%(name,country,chineselevel))  
StudentInfo(country='America', chineselevel='B', name='John')
```

用纳1通过 25ms 4856kb 2

提交答案 本次得分/总分: 2.00/2.00分

第二题

请改正下面程序中存在的错误。

```
def StudentInfo(name, **args):  
    print(name, args)  
StudentInfo('Li Xiaoming', 'China', 'A')
```

改正:

```
def StudentInfo(name, *args):  
    print(name, args)  
StudentInfo('Li Xiaoming', 'China', 'A')
```

2 请改正下面程序中存在的错误。

```
def StudentInfo(name, *args):  
    print(name,args)  
    StudentInfo('Li Xiaoming', 'China', 'A') (2分)
```

题目内容：

请改正下面程序中存在的错误。

```
def StudentInfo(name, *args):  
    print(name,args)  
    StudentInfo('Li Xiaoming', 'China', 'A')
```

输出样例：

Li Xiaoming ('China', 'A')

时间限制：500ms 内存限制：32000kb

Python3.4

```
def StudentInfo(name, *args):  
    print(name,args)  
    StudentInfo('Li Xiaoming', 'China', 'A')
```

用例测试结果	运行时间	占用内存	提示	得分
用例1通过	46ms	4612kb		2

提交答案

本次得分/总分: 2.00/2.00分

第三题

请改正下面程序中存在的错误。

```
def Sum(a, b, c):  
    print(a+b+c)
```

```
t=(1, 2, 3)
```

```
Sum(*t)
```

改正：

```
def Sum(a, b, c):  
    s=a+b+c  
    return s
```

```
t=(1, 2, 3)
```

```
a=Sum(*t)
```

```
print(a)
```

3 请改正下面程序中存在的错误。

```
def Sum(a,b,c):  
    print(a+b+c)  
    t=(1,2,3)  
    Sum("") (2分)
```

题目内容：
请改正下面程序中存在的错误。

```
def Sum(a,b,c):  
    print(a+b+c)  
    t=(1,2,3)  
    Sum("")
```

输出样例：
6

时间限制：500ms 内存限制：32000kb

Python3.4

```
def Sum(a,b,c):  
    s=a+b+c  
    return s  
t=(1,2,3)  
a=Sum(t)  
print(a)
```

用例测试结果	运行时间	占用内存	提示	得分
用例1通过	30ms	4620kb		2

提交答案

本次得分/总分: 2.00/2.00分

第四题

请改正下面程序中存在的错误。

```
def f1():  
    print(x)  
def f2():  
    x=50 #将全局变量 x 的值修改为 50  
    print(x)  
x=10  
f2()  
f1()  
改正:  
def f1():  
    print(x)  
def f2():  
    global x    #global 关键字  
    x=50 #将全局变量 x 的值修改为 50  
    print(x)  
x=10  
f2()  
f1()
```

4 请改正下面程序中存在的错误。

```
def f1():  
    print(x)  
def f2():  
    x=50 #将全局变量x的值修改为50  
    print(x)  
    x=10  
f2()  
f1() (2分)
```

测试内容:

请改正下面程序中存在的错误。

```
def f1():  
    print(x)  
def f2():  
    x=50 #将全局变量x的值修改为50  
    print(x)  
    x=10  
f2()  
f1()
```

输出样例:

50
10

时间限制: 500ms 内存限制: 32000kb

Python3.4

```
def f1():  
    print(x)  
def f2():  
    global x  
    x=50 #将全局变量x的值修改为50  
    print(x)  
    x=10  
f2()  
f1()
```

用例测试通过

运行时间

占用内存

提示

得分

用例1通过

30ms

4816kb

2

提交答案

本次得分/总分: 2.00/2.00分

第五题

请改正下面程序中存在的错误。

```
def deco(func):  
    def inner():  
        print('deco begin')  
        func()  
        print('deco end')  
    return inner  
@deco  
def add(a,b):  
    print(a+b)  
if __name__=='__main__':  
    add(3,5)
```

改正:

```
def deco(func):  
    def inner(x,y): #此处的参数与参数传来的 func 实参的参数个数一致  
        print('deco begin')  
        func(x,y)  
        print('deco end')  
    return inner  
@deco  
def add(a,b):
```

```

    print(a+b)
if __name__ == '__main__':
    add(3,5)

```

5

请改正下面程序中存在的错误。

```

def deco(func):
    def inner():
        print('deco begin')
        func()
        print('deco end')
    return inner
@deco
def add(a,b):
    print(a+b)
if __name__ == '__main__':
    add(3,5) (2分)

```

题目内容：

请改正下面程序中存在错误。

```

def deco(func):
    def inner():
        print('deco begin')
        func()
        print('deco end')
    return inner
@deco
def add(a,b):
    print(a+b)
if __name__ == '__main__':
    add(3,5)

```

输出样例：

```

deco begin
8
deco end

```

时间限制：600ms 内存限制：32000kb

Python3.4

```

def deco(func):
    def inner(x,y):
        print('deco begin')
        func(x,y)
        print('deco end')
    return inner
@deco
def add(a,b):
    print(a+b)
if __name__ == '__main__':
    add(3,5)

```

用例测试名称	运行时间	占用内存	提示	得分
用例1通过	41ms	4016kb		2

提交答案

本次得分/总分: 2.00/2.00分

第六题

编写程序实现以下功能：输入若干整数（输入 0 结束），每个整数输入完毕后，马上输出该整数是否为素数。要求判断一个整数是否为素数的功能用一个函数实现。

```

def isPrime(n):
    if n <= 1:
        return "Invalid"
    num = int(n ** 0.5)
    for i in range(2,num + 1):
        if n % i == 0:
            return "No"
    return "Yes"
res = []
while True:
    n = int(input())
    if n == 0:

```

```

        break
    res.append(isPrime(n))
for i in res:
    print(i)

```

6 编写程序实现以下功能：输入若干整数（输入0结束），每个整数输入完毕后，马上输出该整数是否为素数。要求判断一个整数是否为素数的功能用一个函数实现。（2分）

题目内容：

编写程序实现以下功能：输入若干整数（输入0结束），每个整数输入完毕后，马上输出该整数是否为素数。要求判断一个整数是否为素数的功能用一个函数实现。

输入格式：

若干整数，输入0结束处理。

输出格式：

输入整数小于2时输出Invalid，输入整数是素数时输出Yes，输入整数不是素数时输出No。

输入样例：

```

3
-1
50
1
19
0

```

输出样例：

```

Yes
Invalid
No
Invalid
Yes

```

时间限制：500ms 内存限制：32000kb

Python3.4

```

def isPrime(n):
    if n <= 1:
        return "Invalid"
    num = int(n**0.5)
    for i in range(2,num+1):
        if n%i==0:
            return "No"
    return "Yes"
res = []
while True:
    n = int(input())
    if n == 0:

```

用例测试名称	运行时间	占用内存	提示	得分
用例1通过	63ms	4612kb		1
用例2通过	24ms	4610kb		1

提交答案

本次得分/总分：2.00/2.00分

第七题

编写程序实现以下功能：输入两个字符串，如果第一个字符串是第二个字符串的前缀，则输出第一个字符串；如果第二个字符串是第一个字符串的前缀，则输出第二个字符串；如果两个字符串互相都不为前缀则输出'no'。要求判断一个字符串是否是另一个字符串前缀的功能用函数实现。

```

Str1=input()
Str2=input()
def compare(str1,str2):
    i=0
    n=min(len(str1),len(str2))
    if len(str1)>=len(str2): #把最小字符串给 x
        x=str2
    else:
        x=str1
    while i<=n:
        if i==n:

```

```

        return x
    elif str1[i]==str2[i]:
        i+=1
        continue
    else:
        return 'no'
print(compare(Str1,Str2))

```

7 编写程序实现以下功能：输入两个字符串，如果第一个字符串是第二个字符串的前缀，则输出第一个字符串；如果第二个字符串是第一个字符串的前缀，则输出第二个字符串；如果两个字符串互相都不为前缀则输出'no'。要求判断一个字符串是否是另一个字符串前缀的功能用函数实现。（3分）

题目内容：

编写程序实现以下功能：输入两个字符串，如果第一个字符串是第二个字符串的前缀，则输出第一个字符串；如果第二个字符串是第一个字符串的前缀，则输出第二个字符串；如果两个字符串互相都不为前缀则输出'no'。要求判断一个字符串是否是另一个字符串前缀的功能用函数实现。

输入格式：

两个字符串。

输出格式：

一个字符串是另一个字符串前缀则输出yes，否则输出no。

输入样例：

substring
sub

输出样例：

sub

输入样例：

sub
substring

输出样例：

sub

输入样例：

substring
string

输出样例：

no

时间限制：500ms 内存限制：32000kb

Python3.4

```

Str1=input()
Str2=input()
def compare(str1,str2):
    n=0
    m=min(len(str1),len(str2))
    if len(str1)>len(str2):
        x=str2
    else:
        x=str1
    while n<m:
        if str1[n]==str2[n]:
            n+=1
        else:
            return 'no'
    return 'yes'

```

用例测试结果	运行时间	占用内存	提示	得分
用例1通过	79ms	4624kb		1
用例2通过	29ms	4628kb		1
用例3通过	29ms	4632kb		1

提交答案

本次得分/总分: 3.00/3.00分

第八题

编写程序实现以下功能：计算 $C(1, n) + C(2, n) + \dots + C(m, n)$ 的值。其中， m 和 n 是两个正整数，且 m 小于或等于 n ， $C(i, n) = n! / i! / (n-i)!$ （这里的!表示阶乘， i 在 $1 \sim m$ 上依次取值）。要求计算 $C(i, n)$ 的功能用函数实现。

```

def fac(n):
    if n == 1:
        return 1
    return n * fac(n-1)

```

```
def cin(i,n):
    return fac(n)/(fac(i)*fac(n-i))

m = int(input())
n = int(input())
if m*n <= 0 or m > n:
    print("invalid")
else:
    sum = 0
    for i in range(1,m+1):
        sum += cin(i, n)
    print(int(sum))
```

8 编写程序实现以下功能：计算 $C(1,n)+C(2,n)+\dots+C(m,n)$ 的值。其中， m 和 n 是两个正整数，且 m 小于或等于 n ， $C(i,n)=n!/(i!(n-i)!)$ （这里的 $!$ 表示阶乘， i 在 $1\sim m$ 上依次取值）。要求计算 $C(i,n)$ 时能用函数实现。（2分）

题目内容：

编写程序实现以下功能：计算 $C(1,n)+C(2,n)+\dots+C(m,n)$ 的值。其中， m 和 n 是两个正整数，且 m 小于或等于 n ， $C(i,n)=n!/(i!(n-i)!)$ （这里的 $!$ 表示阶乘， i 在 $1\sim m$ 上依次取值）。要求计算 $C(i,n)$ 的功能用函数实现。

输入格式：

两个正整数 m 和 n ，且 m 小于或等于 n 。

输出格式：

如果输入的两个整数无效，则输出invalid；否则输出计算结果。

输入样例：

1

3

输出样例：

invalid

输入样例：

2

3

输出样例：

6

时间限制：500ms 内存限制：32000kb

Python3.4

```
def fac(n):
    if n == 1:
        return 1
    return n * fac(n-1)

def cin(i,n):
    return fac(n)/(fac(i)*fac(n-i))

m = int(input())
n = int(input())
if m*n <= 0 or m > n:
    print("invalid")
```

用例测试结果	运行时间	占用内存	提示	得分
用例1通过	40ms	4058kb		1
用例2通过	30ms	4024kb		1

提交答案

本次得分/总分：2.00/2.00分

第九题

请编写程序实现以下功能：用户输入两个数据，如果两个数据都是整数（即都由数字组成），则输出其减法运算结果；如果两个数据都不是整数，则将这两个数据作为两个字符串、输出这两个字符串连接的结果；如果一个数据不是整数、另一个数据是整数，则将不是整数的数据作为字符串、输出字符串和整数重复运算的结果。要求判断一个输入数据是否是整数的功能用函数实现。


```
str1=input()
str2=input()

def number(n):
    for i in n:
        if i not in "0123456789":
            return False
    return True

if number(str1) and number(str2):
    print(int(str1)-int(str2))
elif not number(str1) and not number(str2):
    print(str1+str2)
else:
    if number(str1):
        print(int(str1)*str2)
    else:
        print(int(str2)*str1)
```

9 请编写程序实现以下功能：用户输入两个数据，如果两个数据都是整数（即都由数字组成），则输出其加减乘除运算结果；如果两个数据都不是整数，则将这两个数据作为两个字符串，输出这两个字符串连接的结果；如果一个数据不是整数，另一个数据是整数，则将不是整数的数据作为字符串，输出字符串和整数重复运算的结果。要求判断一个输入数据是否是整数的功能用函数实现。（3分）

题目内容：

请编写程序实现以下功能：用户输入两个数据，如果两个数据都是整数（即都由数字组成），则输出其加减乘除运算结果；如果两个数据都不是整数，则将这两个数据作为两个字符串，输出这两个字符串连接的结果；如果一个数据不是整数，另一个数据是整数，则将不是整数的数据作为字符串，输出字符串和整数重复运算的结果。要求判断一个输入数据是否是整数的功能用函数实现。

输入格式：

两个数据

输出格式：

运算结果。

输入样例：

3

5

输出样例：

-2

输入样例：

hello

3

输出样例：

hellohellohello

输入样例：

hello

world

输出样例：

helloworld

时间限制：500ms 内存限制：32000kb

Python3.4

```
str1=input()
str2=input()

def number(n):
    for i in n:
        if not i in "0123456789":
            return False
    return True

if number(str1) and number(str2):
    print((str1)+(str2))
elif not number(str1) and not number(str2):
```

用例测试结果	运行时间	占用内存	提示	得分
用例1通过	28ms	4820kb		1
用例2通过	30ms	4816kb		1
用例3通过	30ms	4832kb		1

提交答案

本次得分/总分：3.00/3.00分

第十题

编写程序解决汉诺塔问题：有三根杆（编号 A、B、C），在 A 杆自下而上、由大到小按顺序放置 n 个盘子（编号从 n 至 1，即最下面盘子编号为 n，最上面盘子编号为 1）。目标：把 A 杆上的盘子全部移到 C 杆上，并仍保持原有顺序叠好。操作规则：每次只能移动一个盘子，并且在移动过程中三根杆上都始终保持大盘在下，小盘在上，操作过程中盘子可以置于 A、B、C 任一杆上。提示：编写递归函数解决该问题。

```
def hanoi(n, A, B, C):
    if n == 1:
        print("{}: {}->{}".format(n, A, C))
    else:
        hanoi(n - 1, A, C, B)
        print("{}: {}->{}".format(n, A, C))
        hanoi(n - 1, B, A, C)
n = int(input())
```

hanoi(n, "A", "B", "C")

10 编写程序解决汉诺塔问题：有三根杆（编号A、B、C），在A杆自下而上、由大到小按顺序放置n个盘子（编号从n至1，即最下面盘子编号为n，最上面盘子编号为1）。目标：把A杆上的盘子全部移到C杆上，并保持初始有序性。操作规则：每次只能移动一个盘子，并且在移动过程中三根杆上都必须保持大盘在下，小盘在上，操作过程中盘子可以置于A、B、C任一杆上。提示：编写递归函数解决该问题。（2分）

题目内容：

编写程序解决汉诺塔问题：有三根杆（编号A、B、C），在A杆自下而上、由大到小按顺序放置n个盘子（编号从n至1，即最下面盘子编号为n，最上面盘子编号为1）。目标：把A杆上的盘子全部移到C杆上，并保持初始有序性。操作规则：每次只能移动一个盘子，并且在移动过程中三根杆上都必须保持大盘在下，小盘在上，操作过程中盘子可以置于A、B、C任一杆上。提示：编写递归函数解决该问题。

输入格式：

盘子个数n。

输出格式：

每一行输出一个盘子移动的操作，格式为：盘子编号-源杆编号->目标杆编号。

输入样例：

3

输出样例：

1 A->C
2 A->B
1 C->B
3 A->C
1 B->A
2 B->C
1 A->C

时间限制：600ms 内存限制：32000kb

Python3.4

```
def hanoi(n, A, B, C):  
    if n == 1:  
        print("{} {}->{}".format(n, A, C))  
    else:  
        hanoi(n - 1, A, C, B)  
        print("{} {}->{}".format(n, A, C))  
        hanoi(n - 1, B, A, C)  
  
n = int(input())  
hanoi(n, "A", "B", "C")
```

用例测试结果	运行时间	占用内存	提示	得分
用例1通过	30ms	4824kb		2

提交答案

本次得分/总分：2.00/2.00分