**R7-1 2.1 计算本金利息的和 (10 分)**

用a=eval(input()),b=eval(input()),n=eval(input())分别输入储户的本金a、年利率b和存款年数n，再计算本金利息的和，计算公式为：a x pow((1+b),n) 要求：用format（）格式化数字，宽度为10，在指定宽度内右对齐，不够用"+"填充，保留2位小数。

输入格式:

本金a、年利率b和存款年数n，各一行。

输出格式:

本金利息的之和。

输入样例:

在这里给出一组输入。例如：

5243

0.03

5

输出样例:

输出格式：

+++6078.07

a = eval(input())

b = eval(input())

n = eval(input())

s = a\*pow((1+b),n)

print("{0:+>10.2f}".format(s))

**R7-2 计算逆序的四位数 (10 分)**

输入一个四位数，得到一个新的四位数。新数的千位数字、百位数字、十位数字和个位数字分别是原数的个位数、十位数、百位数和千位数。

输入格式:

输入一个四位数（假设输入必定是四位数，并且最后一位不为0）。

输出格式:

输出一个四位数。

输入样例:

在这里给出一组输入。例如：

1234

输出样例:

在这里给出相应的输出。例如：

4321

s = input()

a = s[3]+s[2]+s[1]+s[0]

print(a)

**R7-3 学号切片 (10 分)**

输入自己的学号，然后利用字符串切片分解并输出年级、专业、学号。

输入格式:

学号为字符串

输出格式:

在不同行上输出年级代码、专业代码、学号

输入样例:

在这里给出一组输入。例如：

2019244001

输出样例:

在这里给出相应的输出。例如：

2019

244

001

a = input()

b = (a[0]+a[1]+a[2]+a[3])

c = (a[4]+a[5]+a[6])

d = (a[7]+a[8]+a[9])

print(b+" ")

print(" "+c+" ")

print(" "+d)

**R7-1 3.2 判断一个数是奇数还是偶数 (20 分)**

用x=eval(inpu())输入一个正整数，然后判断它是奇数还是偶数，但无论是奇数还是偶数最后都要将判断结果按题后的格式输出，"even number"表示偶数，"odd number"表示奇数，本题的输出控制语句举例 ：print(x,"even number")。

输入格式:

输入一个正整数。

输出格式:

将判断结果按样例的格式输出，本题的输出控制语句举例 ：print(x,"even number")。

输入样例1:

输入15：

15

输出样例1:

相应的输出：

15 odd number

输入样例2:

输入10：

10

输出样例2:

相应的输出：

10 even number

x = eval(input())

if (x % 2)==0:

print(x,"even number")

else:

print(x,"odd number")

**R7-2 3.3 购物打折 (25 分)**

某商场春节期间推出购物打折优惠活动：

一次性购物1000元以下（不含1000）不打折；

1000元以上（含1000）到2000元以下（不含2000）9.5折；

2000元以上（含2000）到3000元以下（不含3000）8.5折；

3000元以上（含3000）到4000元以下（不含4000）7.5折；

4000元以上（含4000）6.5折。

请编写一个程序，根据x=eval(inpu())语句输入的购物金额计算消费者实际应付金额，结果保留2位小数，并将结果按题后规定的格式输出，“after discount“表示打折后。

提示： 结果保留2位小数既可以用“%”来格式化，也可以使用format（）方法格式化.格式错误会没有得分。

输入格式:

输入的购物金额。

输出格式:

计算实际应付金额，结果保留2位小数，本题的输出格式控制语句：print("after discount:","%.2f"%s)。

输入样例1:

在这里给出一组输入6000：

6000

输出样例1:

相应的输出：

after discount: 3900.00

输入样例2:

在这里给出一组输入854：

854

输出样例2:

相应的输出：

after discount: 854.00

x = eval(input())

if x < 1000:

print("after discount: {:.2f}".format(x))

elif 1000 <= x <2000:

print("after discount: {:.2f}".format(x\*0.95))

elif 2000 <= x <3000:

print("after discount: {:.2f}".format(x\*0.85))

elif 3000 <= x <4000:

print("after discount: {:.2f}".format(x\*0.75))

else:

print("after discount: {:.2f}".format(x\*0.65))

**R7-3 3.1 2个数x和y，让x大于等于y (20 分)**

1运用x=eval(inpu()),y=eval(inpu())输入2个数分别赋值给x和y并输出x、y的值，再比较x和y的值，如果y>x，就把x和y的值交换最后再重新输出x、y的值，按题目后要求的格式输出，本题的输出格式控制语句举例：print("Before exchange:","x=",x,"y=",y)。

输入格式:

输入2个数。

输出格式:

输出格式按样例，本题的输出格式控制语句举例：print("Before exchange:","x=",x,"y=",y)。

输入样例:

在这里给出一组输入。例如：

51

67

输出样例:

相应的输出：

Before exchange: x= 51 y= 67

After exchange: x= 67 y= 51

x = eval(input())

y = eval(input())

print("Before exchange:","x=",x,"y=",y)

if y > x:

x,y = y,x

print("After exchange:","x=",x,"y=",y)

else:

print("After exchange:","x=",x,"y=",y)

**R7-1 jmu-python-汇率兑换 (10 分)**

按照1美元=6人民币的汇率编写一个美元和人民币的双向兑换程序

输入格式:

输入人民币或美元的金额，人民币格式如：R100，美元格式如：$100

输出格式:

输出经过汇率计算的美元或人民币的金额，格式与输入一样，币种在前，金额在后，结果保留两位小数

输入样例1:

R60

输出样例1:

$10.00

输入样例2:

$5

输出样例2:

R30.00

money = input()

unit = money[0]

if money[0] == 'R':

U=eval(money[1:])/6

print("$%.2f" % U)

elif money[0] == '$':

R=6\*eval(money[1:])

print("R%.2f" % R)

**R7-2 jmu-python-成绩转换 (10 分)**

本题要求编写程序将一个百分制成绩转换为五分制成绩。转换规则：

大于等于90分为A； 小于90且大于等于80为B； 小于80且大于等于70为C； 小于70且大于等于60为D； 小于60为E。

输入样例:

98

输出样例:

A

x = int(input())

if x >= 90:

y = 'A'

elif 80 <= x <90:

y = 'B'

elif 70 <= x <80:

y = 'C'

elif 60 <= x <70:

y = 'D'

elif x < 60:

y = 'E'

print(y)

**R7-3 3.4 验证登录用户名与密码程序 (24 分)**

请你编写一个用户登录程序：先用u = input()输入用户名给u，再用p = input()输入用户密码给p。

然后判断输入的用户名是不是“user”，如果不是，则输出“Sorry! The user name your entered does not exist!”，结束程序；

如果输入的用户名是“user”则进一步判断输入的密码是不是正确密码“666666”，如果是这个正确密码，则输出“Login was successful！”，如果密码不是这个正确密码，则输出“Password error!”，结束程序。

程序运行结果按规定的格式输出。

输入格式:

输入用户名和密码。

输出格式:

按规定的格式输出。

输入样例1:

输入用户名user. 密码666666：

user

666666

输出样例1:

相应的输出：

Login was successful!

输入样例2:

输入用户名user. 密码666：

user

666

输出样例2:

相应的输出：

Password error!

输入样例3:

输入用户名user. 密码666666：

use

666666

输出样例3:

相应的输出：

Sorry! The user name your entered does not exist!

u = input()

p = input()

if u == "user":

if p == "666666":

print("Login was successful!")

else:

print("Password error!")

else:

print("Sorry! The user name your entered does not exist!")

**R7-4 重复打印 (10 分)**

要求重复打印字符或文字

输入格式:

如输入“x个y”

输出格式:

重复打印y字符，次数为x次

输入样例:

在这里给出一组输入。例如：

3个A

输出样例:

在这里给出相应的输出。例如：

AAA

输入样例:

在这里给出一组输入。例如：

5个人

输出样例:

在这里给出相应的输出。例如：

人人人人人

input = input().split("个")

x = int(input[0])

y = input[1]

print(y \* x)

**R7-1 猴子吃桃 (10 分)**

猴子吃桃问题。猴子第一天摘下若干个桃子，当即吃了一半，还不过瘾，又多吃了一个。第二天早上又将剩下的桃子吃掉一半，又多吃了一个。以后每天早上都吃了前一天剩下的一半零一个。到第n(n从键盘输入)天早上想再吃时，就只剩下一个桃子了。求第一天共摘了多少个桃子(不考虑猴子是否真的能吃多少桃子)。

输入格式:

输入整型天数n

输出格式:

输入样例:

在这里给出一组输入。例如：

5

输出样例:

在这里给出相应的输出。例如：

Day=5 Total=46

a = 1

n =eval(input())

for i in range (n,1,-1):

a = (a+1)\*2

print("Day={} Total={}".format(n,a))

**R7-2 if-else双分支计算行李托运费 (10 分)**

铁路托运行李规定，行李重量不超过50公斤的（含50公斤），托运费按每公斤0.25元计费；如超过50公斤，超过的部分每公斤加收0.2元。输入行李的重量x，计算并输出应付的托运费y。

输入格式:

输出格式:

结果打印2位小数

输入样例1:

在这里给出一组输入。例如：

37

输出样例1:

在这里给出相应的输出。例如：

x=37.00 y=9.25

输入样例2:

在这里给出一组输入。例如：

75

输出样例2:

在这里给出相应的输出。例如：

x=75.00 y=23.75

x = eval(input())

if x <= 50:

y = 0.25\*x

print("x={:.2f} y={:.2f}".format(x,y))

else:

y = 0.25\*50+(x-50)\*0.45

print("x={:.2f} y={:.2f}".format(x,y))

**R7-3 4.2 while循环计算偶数和 (20 分)**

请用n=eval(input())输入一个大于100的偶数，用while循环求小于等于n范围内的所有偶数和，并按题后规定的格式输出计算结果，本题输出格式控制语句 "2+4+6+...+{0}={1}".format(n,s)。

输入格式:

输入一个大于100的偶数。

输出格式:

本题输出格式控制语句"2+4+6+...+{0}={1}".format(n,s)，例如输入200，输出为：2+4+6+...+200=10100

输入样例:

输入200：

200

输出样例:

相应的输出：

2+4+6+...+200=10100

n = eval(input())

a = 0

i = 0

while i <= n:

a = a+i

i = i+2

print("2+4+6+...+{0}={1}".format(n,a))

**R7-4 4.3 运用while循环处理学生成绩 (25 分)**

3请先用n=eval(input()）输入学生人数n，作为while的循环条件，再在while循环用s = int(input())输入这n个同学成绩，循环一次输入一个同学的成绩，求出这些同学成绩平均值，最大值和最小值。程序运行结果按题后规定的格式输出，依次输出成绩平均值、最大值和最小值，成绩平均值保留一位小数，其他为整数，本题的输出格式控制语句举例：print("max:{0:}".format(s\_max)) （输出最高分）。

输入格式:

输入学生人数，输入这n个同学成绩。

输出格式:

依次输出成绩平均值、最大值和最小值，成绩平均值保留一位小数(用format（）控制小数位数），其他为整数，本题的输出格式控制语句举例：print("max:{0:}".format(s\_max)) （输出最高分），平均分，最小值的控制格式一样。，注意输出顺序！

输入样例:

给出一组输入，5是学生人数，下面是输入的5个学生成绩：

5

93

87

98

65

71

输出样例:

相应的输出：

average:82.8

max:98

min:65

n = eval(input())

s = 0

maxx = 0

minxx = 1000

for i in range(n):

a = eval(input())

s += a

if a > maxx:

maxx = a

if a < minxx:

minxx = a

s = s/n

print("average:{0:.1f}".format(s))

print("max:{0:}".format(maxx))

print("min:{0:}".format(minxx))

**R7-5 4.4 for循环统计各种字符个数 (25 分)**

请编写一个运用for循环来分别统计输入的字符串中有多少个大写英文字母（用isupper()方法判断）、小写英文字母（用islower() 方法判断）、数字（用isdigit ()方法判断）、空格（用isspace ()方法判断）和其他字符。程序进入循环之前先用input（）输入要统计的字符串。程序运行结果按题后规定的格式输出。依次输出大写字母、小写字母、数字、空格、其他字符的个数，本题输出格式控制举例：print ("upper:",n1) （输出大写字母个数），其他的控制格式语法一样。注意输出顺序！

提示：字符串的相关方法的使用见教材162页。本题循环体内利用多分支语句就可以分别判断各种字符。

输入格式:

输入要统计的字符串。

输出格式:

依次输出大写字母、小写字母、数字、空格、其他字符的个数，本题输出格式控制举例：print ("upper:",n1) （输出大写字母个数）

输入样例:

在这里给出一组输入。例如：

AcdfKJ3!3&

输出样例:

相应的输出：

upper: 3

lower: 3

digit: 2

space: 0

other: 2

data = list(input())

upper = 0

lower = 0

digit = 0

space = 0

other = 0

for i in data:

if i.isupper():

upper = upper+1

elif i.islower():

lower = lower+1

elif i.isspace():

space = space+1

elif i.isdigit():

digit = digit+1

else:

other = other+1

print("upper:",upper)

print("lower:",lower)

print("digit:",digit)

print("space:",space)

print("other:",other)

**R7-1 打印菱形图形 (10 分)**

输入任意整数n，打印菱形图形。图形样式参见输出格式。

输入格式:

输入一个整数存入n

输出格式:

打印菱形

输入样例:

在这里给出一组输入。例如：

3

输出样例:

在这里给出相应的输出。例如：

\*

\*\*\*

\*\*\*\*\*

\*\*\*

\*

n = int(input())

for i in range(1,2 \* n):

space\_count = 1

star\_count = 0

if i <= n:

space\_count = n - i + 1

else:

space\_count = i - n + 1

if i <= n:

star\_count = 2 \* i - 1

else:

star\_count = 2 \* (2 \* n - i) - 1

for \_ in range(space\_count):

print(" ",end="")

for \_ in range(star\_count):

print("\*",end="")

print()

**R7-2 统计字符数 (10 分)**

输入任意字符串，分别统计并输出其中的大写字母、小写字母、数字及其它符号个数。

输入格式:

输入任意字符串

输出格式:

输出格式见输出样例

输入样例:

在这里给出一组输入。例如：

sf34$^\*YRy56

输出样例:

在这里给出相应的输出。例如：

Uppercase=2 lowercase=3 number=4 Other=3

sentence = input()

uppercase = 0

lowercase = 0

number = 0

other = 0

for ch in sentence:

if ch.isupper():

uppercase += 1

elif ch.islower():

lowercase += 1

elif ch.isnumeric():

number += 1

else:

other += 1

print("Uppercase=%d lowercase=%d number=%d Other=%d"%(uppercase,lowercase,number,other))

**R7-3 公司营销提成 (5 分)**

某公司营销提成办法：月营销总额在5万（含5万）以内不提成；月营销总额在5万到10万（含10万）提成10%；月营销总额在10万以上提成20%。月营销总额由语句x=eval(input())输入并赋值，根据X计算提成金额并输出，结果保留2位小数，用语句print("{0:.2f}".format(y))控制小数位数），输入输出格式见样例

输入格式:

在一行中输入一个月营销总额的正整数，不含有其他字符。

输出格式:

在一行中输出提成金额，保留2位小数，不含有其他字符。

输入样例:

例如：输入75000的格式为

75000

输出样例:

对应75000计算的提成输出格式为：

7500.00

x = eval(input())

t = 0

if x <= 50000:

t = 0

elif 50000 < x <= 100000:

t = x\*0.1

else:

t = x\*0.2

print("{0:.2f}".format(t))

在这里给出相应的输出。例如：

no

### 输入样例2:

在这里给出一组输入。例如：

2012

### 输出样例2:

在这里给出相应的输出。例如：

yes

n = eval(input())

if n % 400 == 0 or (n % 4 == 0 and n % 100 != 0):

print("yes")

else:

print("no")

**R7-5 利用遍历循环求1-2+3-4+5 ... n的所有数的和 (5 分)**

通过语句n=eval(input()) 输入一个数并赋值n，利用遍历循环求1-2+3-4+5 ... n的所有数的和，注意偶数是减法。

输入格式:

输入一个正整数。

输出格式:

输出求和结果。

输入样例:

输入一个正整数。例如：

99

输出样例:

输出求和结果。例如：

50

n = eval(input())

s = 0

for i in range(n+1):

s = s + i\*(-1)\*\*(i+1)

print(s)

**R7-1 求小于等于给定整数范围内的所有质数的和 (10 分)**

用m=eval(input())语句输入一个正整数并赋值给m,求小于等于m的所有质数的和，输出程序计算结果。

输入格式:

输入一个正整数。

输出格式:

输出求和结果。

输入样例1:

在这里给出一组输入。例如：

10

输出样例1:

样例1的质数和的输出格式如下：

17

输入样例2:

在这里给出一组输入。例如：

11

输出样例2:

样例2的质数和的输出格式如下：

28

m = eval(input())

sum = 0

for i in range(2,m+1):

for j in range(2,i):

if i % j == 0:

break

else:

sum += i

print(sum)

**R7-2 5.4 for循环嵌套打印直角三角形 (10 分)**

请用for循环嵌套来编写打印直角三角形的程序；进入循环之前用n=int(input())语句输入要打印的行数。程序运行结果按题后规定的格式输出。提示：有多少行外循环就要循环多少次，而内循环次数就是每行打印的字符数量，图形由”$”构成。

输入格式:

输入要打印的行数。

输出格式:

每行输出字符数按图形要求输出

输入样例:

输入行数8：

5

输出样例:

相应的输出：

$

$$

$$$

$$$$

$$$$$

n = int(input())

for i in range(1,n+1):

if i == n:

print('$'\*i,end="")

else:

print('$'\*i)

**R7-3 循环嵌套 (10 分)**

某商店有小鸡玩偶出售，标价是公鸡5元/只，母鸡2元/只，小鸡1元/只， 请设计程序实现：当用户有n元准备购买小鸡玩偶时，如果每种小鸡至少购买一只，输出购买三种玩偶所有可能的数量，格式输出参照输出样例。

输入格式:

直接输入购买金额。

输出格式:

依次输出购买的每种类型的小鸡数量，要求：

依次输出顺序为：公鸡数量，母鸡数量，小鸡数量，并按照公鸡和母鸡玩偶从小到大输出递增，请使用format()函数，如："公鸡= 只,母鸡= 只,小鸡=只"。

请注意观察参考两个输出样例

输入样例:

10

输出样例:

在这里给出相应的输出。例如：

公鸡=1只,母鸡=1只,小鸡=3只

公鸡=1只,母鸡=2只,小鸡=1只

输入样例:

15

输出样例:

在这里给出相应的输出。例如：

公鸡=1只,母鸡=1只,小鸡=8只

公鸡=1只,母鸡=2只,小鸡=6只

公鸡=1只,母鸡=3只,小鸡=4只

公鸡=1只,母鸡=4只,小鸡=2只

公鸡=2只,母鸡=1只,小鸡=3只

公鸡=2只,母鸡=2只,小鸡=1只

n = eval(input())

for i in range(1,n//5):

for a in range(1,n//2):

for b in range(1,n):

if (5\*i + 2\*a +1\*b)==n:

print("公鸡="+str(i)+"只,母鸡="+str(a)+"只,小鸡="+str(b)+"只")

**R7-1 输出字母在字符串中位置索引 (10 分)**

输入一个字符串，再输入两个字符，求这两个字符在字符串中的索引。

输入格式:

第一行输入字符串  
第二行输入两个字符，用空格分开。

输出格式:

从右向左输出字符和索引，即下标最大的字符最先输出。每行一个。

输入样例:

在这里给出一组输入。例如：

pmispsissippi

s p

输出样例:

在这里给出相应的输出。例如：

11 p

10 p

8 s

7 s

5 s

4 p

3 s

0 p

n = input()

a,b = input().split()

n = n[::-1]

for i in range (0,len(n)):

if n[i] == a:

print("%d %c" %(len(n)-i-1,a))

elif n[i] == b:

print("%d %c" %(len(n)-i-1,b))

**R7-2 打印九九乘法表 (10 分)**

利用循环嵌套编写代码打印出与输出样例一样的九九乘法表；要求能根据用户输入1~9的某个自然数n，编写程序输出该自然数（最大值不超过该自然数n）以内的九九乘法表，具体格式请分析下方输出样例。（图中乘法符号用小写字母x表示，乘积结果占两个整数的宽度，可使用%2d或者{:2d}）

输入格式:

输入1-9的任意自然数

输出格式:

按照图示结构输出九九乘法表

输入样例:

例如：

3

输出样例:

输出：

1x1= 1

1x2= 2 2x2= 4

1x3= 3 2x3= 6 3x3= 9

输入样例:

例如：

9

输出样例:

输出：

1x1= 1

1x2= 2 2x2= 4

1x3= 3 2x3= 6 3x3= 9

1x4= 4 2x4= 8 3x4=12 4x4=16

1x5= 5 2x5=10 3x5=15 4x5=20 5x5=25

1x6= 6 2x6=12 3x6=18 4x6=24 5x6=30 6x6=36

1x7= 7 2x7=14 3x7=21 4x7=28 5x7=35 6x7=42 7x7=49

1x8= 8 2x8=16 3x8=24 4x8=32 5x8=40 6x8=48 7x8=56 8x8=64

1x9= 9 2x9=18 3x9=27 4x9=36 5x9=45 6x9=54 7x9=63 8x9=72 9x9=81

x = eval(input())

for i in range(1,x+1):

for j in range(1,i+1):

print("{}x{}={:2d}".format(j,i,i\*j),end=" ")

print()

**R7-3 jmu\_python\_最大公约数&最小公倍数 (10 分)**

本题要求从键盘输入两个整数（以逗号间隔），编程求出这两个数的最大公约数和最小公倍数

提示：求最大公约数可用辗转相除法，最小公倍数用两数的积除以最大公约数

输入格式:

在一行中输入两个整数，以逗号间隔

输出格式:

输出“GCD:a, LCM:b"，其中a为求出的最大公约数，b为求出的最小公倍数

**注意：在逗号后面有个空格**

输入样例:

12,14

输出样例:

GCD:2, LCM:84

num1,num2 = input().split(',')

num1 = int(num1)

num2 = int(num2)

a\_num1 = num1

a\_num2 = num2

while a\_num1 != a\_num2:

if a\_num1 < a\_num2:

a\_num1,a\_num2 = a\_num2,a\_num1

a\_num1 = a\_num1 - a\_num2

print("GCD:%d, LCM:%d" % (a\_num1,num1\*num2 // a\_num1))

**R7-1 求“厉害”数 (10 分)**

所谓“厉害”数，是指其各位数字的5次方等于该数本身。编程输出所有长度不超过n位的正整数中的“厉害”数。 例如：4151=4​5​​+1​5​​+5​5​​+1​5​​

输入格式:

一个整数n （1<=n<=6）

输出格式:

每行一个“厉害”数

输入样例:

4

输出样例:

1

4150

4151

n = eval(input())

i = 1

while i < 10\*\*n:

b = i

sum = 0

while b != 0:

a = b % 10

sum += a\*\*5

b //= 10

if sum == i:

print(sum)

i += 1

**R7-2 求所有N位的素数和 (10 分)**

输入一个正整数n(n>=2)，输出所有n位的素数和，如n=2，即输出的是10-99之间的所有素数的和。

输入格式:

一个正整数n(n>=2)

输出格式:

输出所有n位的素数和

输入样例:

2

输出样例:

10-99之间所有的素数和=1043

import math

n=int(input())

front=10\*\*(n-1)

rear=10\*\*(n)

sum=0

for i in range(front,rear):

flag=1

for j in range(2,int(math.sqrt(i))+1):

if i%j==0:

flag=0

break

if flag==1:

sum = sum+i

print("%d-%d之间所有的素数和=%d"%(front,rear-1,sum))

**R7-3 本题要求两个给定正整数的最大公约数和最小公倍数。 (15 分)**

本题要求两个给定正整数的最大公约数和最小公倍数。

输入格式:

输入在两行中分别输入正整数x和y。

输出格式:

在一行中输出最大公约数和最小公倍数的值。

输入样例1:

在这里给出一组输入。例如：

100

1520

输出样例1:

在这里给出相应的输出。例如：

20 7600

x = int(input())

y = int(input())

n = min(x,y)

s = 1

for i in range(1,int(n+1)):

if x%i == 0 and y%i == 0:

s = i

m = x\*y / s

print("%d %d"%(s,m))

**R7-1 列表去重 (10 分)**

输入一个列表，去掉列表中重复的数字，按原来次序输出！

输入格式:

在一行中输入列表

输出格式:

在一行中输出不重复列表元素

输入样例:

在这里给出一组输入。例如：

[4,7,5,6,8,6,9,5]

输出样例:

在这里给出相应的输出。例如：

4 7 5 6 8 9

x = eval(input())

y = list()

for i in x:

if i not in y:

y += [i]

for j in range(0,len(y)-1):

print(y[j],end=" ")

print(y[-1])

**R7-2 删除列表中的重复值（高教社，《Python编程基础及应用》习题4-8） (10 分)**

输入一个列表，删除其中的重复值，再输出。

要求

假设列表中存在k个值为a的元素，删除前k-1个元素，保留最后一个。 不同元素在列表中的相对位置不应被改变。

输入格式:

[元素1， 元素2, ... , 元素n]

输出格式:

[元素1，元素2, ... , 元素k]

输入样例:

[4,3,2,3,2,4,True]

输出样例:

[3, 2, 4, True]

提示：将形如"[1,3,5]"的字符串转换成列表可以使用eval()函数。 注意，输出格式应与输出样例一致，涉及空格，逗号等。

x = eval(input())

y = []

for i in x[::-1]:

if i not in y:

y.append(i)

print(y[::-1])

**R7-3 字符替换 (30 分)**

本题要求编写程序，将输入字符串中的大写英文字母按以下对应规则替换，其他字符不变。(提示：转换表用元组实现）

| **原字母** | **对应字母** |
| --- | --- |
| A | Z |
| B | Y |
| C | X |
| D | W |
| ... | ... |
| X | C |
| Y | B |
| Z | A |

输入格式:

在一行中输入字符串

输出格式:

在一行中给出替换完成后的字符串。

输入样例:

在这里给出一组输入。例如：

This is a pen.

输出样例:

在这里给出相应的输出。例如：

Ghis is a pen.

alpha1 = "abcdefghijklmnopqrstuvwxyz"

alpha2 = "zyxwvutsrqponmlkjihgfedcba"

sentence = input()

for ch in sentence:

if not ch.isalpha() or ch.islower():

print(ch,end="")

continue

new\_ch = alpha2[alpha1.index(ch.lower())]

print(new\_ch.upper(),end="")

**公告**

参考说明：

* Python的绝对值函数为abs(数据)：如abs(-1)结果为1.
* n阶矩阵是指：n x n的矩阵，即n行n列的数据
* 一维列表是指如[ , ,... ] 结构的列表 ,二维列表是指如[ [],[],[]... ]结构的列表

收起

[返回](https://pintia.cn/problem-sets/1382214982664032256/problems/type/7)

**R7-1 列表-统计字母数字符号个数按降序排序 (10 分)**

从键盘输入任意字符串建立列表，分别统计列表中字母、数字、其它符号个数，按降序打印所有字符，打印统计结果。

### 输入格式:

### 输出格式:

### 输入样例:

在这里给出一组输入。例如：

fAhDT35&\*(

### 输出样例:

在这里给出相应的输出。例如：

['h', 'f', 'T', 'D', 'A', '5', '3', '\*', '(', '&']

Letter=5 Number=2 Other=3

x = list(str(input()))

Letter = 0

Number = 0

Other = 0

for i in x:

if i.isalpha():

Letter += 1

elif i.isdigit():

Number += 1

else:

Other += 1

print(sorted(x,reverse=True))

print("Letter={} Number={} Other={}".format(Letter,Number,Other))

**R7-2 求出歌手的得分 (10 分)**

输入一个正整数n (n>4)，再输入n个实数，求出歌手的得分（保留2位小数）。设一歌唱评奖晚会上有n(n>4)个评委为歌手打分.评分规则:每个评委依次打分,再去掉2个最高分和2个最低分,计算余下的分数平均值为歌手的得分.

### 输入格式:

在第一行中输入n 在第二行中输入n个分数

### 输出格式:

在一行中输出平均分数

### 输入样例:

在这里给出一组输入。例如：

10

10 10 9 9 9 8 8 8 7 7

### 输出样例:

在这里给出相应的输出。例如：

aver=8.50

n = int(input())

ls = [int(n) for n in input().split()]

s = 0

ls.sort()

for i in range(0,4):

x = ls.pop(-1)

ls.reverse()

for j in range(len(ls)):

s += ls[j]

li = s / len(ls)

print("aver={:.2f}".format(li))

**R7-3 计算各对应位乘积之和 (10 分)**

读入两个整数a和b，输出绝对值a和绝对值b的各对应位乘积之和，如a=1234，b=608，则输出值为：“1×0+2×6+3×0+4×8“的值，即44。

### 输入格式:

在一行中输入两个数

### 输出格式:

在一行中输出对应位乘积之和

### 输入样例:

在这里给出一组输入。例如：

1234 608

### 输出样例:

在这里给出相应的输出。例如：

44

a,b=map(int,input().split())

a=abs(a)

b=abs(b)

sum=0

while True:

if a==0 and b==0:

break

sum=sum+((a%10)\*(b%10))

a=a//10

b=b//10

print(sum)

**R7-4 求完数 (10 分)**

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。从键盘输入一个正整数(约定该数≤32767、此时因子数≤100)，找出该数以内的所有完数及其因子。

### 输入格式:

从键盘输入一个正整数

### 输出格式:

输出完数，完数的因子用列表输出

### 输入样例:

在这里给出一组输入。例如：

100

### 输出样例:

在这里给出相应的输出。例如：

6 [1, 2, 3]

28 [1, 2, 4, 7, 14]

n = int(input())

for num in range(2, n+1):

l = []

for i in range(1, num):

if (num % i) == 0:

l.append(i)

if sum(l) == num:

print("{} {}".format(num,l))

**R7-5 输出一个 n 阶的方阵 (10 分)**

读入 1 个正整数 n(3≤n<9)和 1 个整数 digit（2<=digit<=9）, 请输出一个 n 阶的方阵, 该矩阵所有边上的元素都是 digit, 其它元素都是 digit-1.

### 输入格式:

在一行中输入n和digt

### 输出格式:

输出n 阶的方阵

### 输入样例:

在这里给出一组输入。例如：

4 2

### 输出样例:

在这里给出相应的输出。例如：

2 2 2 2

2 1 1 2

2 1 1 2

2 2 2 2

a = input().split()

num = int(a[0])

for i in range(num):

print(a[1],end=' ')

print('')

for i in range(1,num-1):

print(a[1],end=' ')

for i in range(1,num-1):

print(int(a[1])-1,end=' ')

print(a[1],end=' ')

print('')

for i in range(num):

print(a[1],end=' ')

print('')

**R7-1 字典-学生成绩计算与查找删除 (10 分)**

输入4个学生的学号、平时成绩、考试成绩，要求完成：(1)计算该学生的总评成绩（平时30%+考试70%）；(2)将学生数据添加到字典（学号为键，平时、考试、总评成绩以列表形式作为值）；(3)输入任意一个学号，在字典中查找该学生，若找到则打印该学生的学号及成绩并删除该学生，打印字典所有元素值；若查找失败，则打印"not find!"。注意：学生的平时、考试、总评成绩均为整型。

### 输入格式:

### 输出格式:

### 输入样例1:

在这里给出一组输入。例如：

1,55,77

2,89,66

3,85,77

4,85,89

3

### 输出样例1:

在这里给出相应的输出。例如：

3 [85, 77, 79]

{'1': [55, 77, 70], '2': [89, 66, 72], '4': [85, 89, 87]}

### 输入样例2:

在这里给出一组输入。例如：

1,77,99

2,87,56

3,87,44

4,80,85

6

### 输出样例2:

在这里给出相应的输出。例如：

not find!

xdict = dict()

for i in range(4):

y = input().split(",")

xsum = eval(y[1])\*0.3 + eval(y[2])\*0.7

xdict[y[0]]=[eval(y[1]),eval(y[2]),int(xsum)]

findx = input()

if findx not in xdict:

print("not find!")

else:

print(findx,xdict[findx])

xdict.pop(findx)

print(xdict)

**R7-2 输出星期名缩写 (10 分)**

输入一个1到7的数字，输出对应的星期名的缩写。  
1 Mon  
2 Tue  
3 Wed  
4 Thu  
5 Fri  
6 Sat  
7 Sun

### 输入格式:

输入1到7之间数字

### 输出格式:

输出对应的星期名的缩写

### 输入样例:

在这里给出一组输入。例如：

1

### 输出样例:

在这里给出相应的输出。例如：

Mon

x = int(input())

weeks = {1:"Mon",

2:"Tue",

3:"Wed",

4:"Thu",

5:"Fri",

6:"Sat",

7:"Sun"}

print(weeks[x])

**R7-3 字典合并与排序 (10 分)**

用户分两行输入两个字典，格式参照输入要求；请设计程序实现：

将两个字典合并为一个字典，如果两个字典中分别有相同的键(key)，需将对应的值相加后作为字典中该键对应的新值。

### 输入格式:

用两行输入字典格式的两个数据，字典中键的数据类型统一为数字或字符，字典中值的数据均为数字。

### 输出格式:

输出合并后的字典数据，字典数据要求按照键的增序排列。

### 输入样例1:

输入1：

{1:2,3:9,5:2,6:2}

{2:1,6:3,7:9,1:4}

### 输出样例1:

输出1：

{1: 6, 2: 1, 3: 9, 5: 2, 6: 5, 7: 9}

### 输入样例2:

输入2：

{'c':3,'a':4,'n':1}

{'a':3,'c':7,'b':10}

### 输出样例2:

输出2：

{'a': 7, 'b': 10, 'c': 10, 'n': 1}

adict = eval(input())

bdict = eval(input())

for key,value in adict.items():

if key in bdict:

bdict[key] += value

else:

bdict[key] = value

bdict = sorted(zip(bdict.keys(),bdict.values()))

d={}

for i in range(len(bdict)):

d[bdict[i][0]]=bdict[i][1]

print(d)

**R7-4 集合和列表，按从小到大的顺序组合成一个字典 (10 分)**

有二个长度一样的集合和列表，按从小到大的顺序组合成一个字典。

### 输入格式:

输入二个长度一样的集合和列表，集合元素全部为整型，列表元素全部为字符串型。

### 输出格式:

组合后的字典。

### 输入样例:

在这里给出一组输入。例如：

{1,2,6,5,3,4}

['aa','cc','bb','d','f','e']

### 输出样例:

在这里给出相应的输出。例如：

{1: 'aa', 2: 'bb', 3: 'cc', 4: 'd', 5: 'e', 6: 'f'}

x = eval(input())

y = eval(input())

X = sorted(x)

Y = sorted(y)

z = dict(zip(X,Y))

print(z)

**R7-5 统计字符串列表中每个字母出现的次数（高教社，《Python编程基础及应用》习题6-8） (10 分)**

统计字符串列表中每个字母出现的次数。  
编写程序，使用eval()函数读入一个仅包含字符串对象的列表，然后统计该列表中每个字母出现的次数。 列表中的字符串对象仅包含小写英文字母。

### 输入格式:

一个仅包括字符串对象的列表，且全部字符串对象中仅出现小写英文字母。

### 输出格式:

字母,次数  
...  
字母,次数  
(注意按a-z的顺序输出)

### 输入样例:

["aaab", "cccdz"]

### 输出样例:

在这里给出相应的输出。例如：

a,3

b,1

c,3

d,1

z,1

x = eval(input())

str1 = "".join(x)

str1 = list(str1)

str1.sort()

n={}

for i in str1:

n[i] = str1.count(i)

for i in n:

print(i+","+str(n[i]))

**R7-1 字典的应用 (10 分)**

先输入5组商品数据（品名，价格）建立字典，然后再输入一种商品数据（品名，价格），在字典中查找最后输入的商品，如果在字典中能找到该商品则打印“Found”并修改其价格为新的价格，否则打印“Not found”并将该商品的品名及价格加入字典，最后打印字典中所有商品的品名及价格。

### 输入格式:

### 输出格式:

### 输入样例:

在这里给出一组输入。例如：

Tea,220

Coffee,150

cola,8.5

Soda,6.4

Beer,12

Brandy,50

### 输出样例:

在这里给出相应的输出。例如：

Not found

Tea 220.0

Coffee 150.0

cola 8.5

Soda 6.4

Beer 12.0

Brandy 50.0

xdict = dict()

for i in range(5):

y = input().split(",")

xdict[y[0]] = y[1]

m = input().split(",")

if m[0] in xdict:

print("Found")

xdict[m[0]] = m[1]

else:

print("Not found")

xdict[m[0]] = m[1]

for j in xdict:

n = eval(xdict[j])

print("{} {:0.1f}".format(j,n))

**R7-2 jmu-Java&Python-统计文字中的单词数量并按出现次数排序 (10 分)**

现在需要统计若干段文字(英文)中的单词数量，并且还需**统计每个单词出现的次数**。

**注1：**单词之间以空格(1个或多个空格)为间隔。  
**注2：**忽略空行或者空格行。

**基本版:**  
统计时，区分字母大小写，且不删除指定标点符号。

**进阶版:**

1. 统计前，需要从文字中删除指定标点符号!.,:\*?。 注意：所谓的删除，就是用1个空格替换掉相应字符。
2. 统计单词时需要忽略单词的大小写。

###输入说明 若干行英文，最后以!!!!!为结束。

###输出说明 单词数量  
出现次数排名前10的单词（次数按照降序排序，如果次数相同，则按照键值的字母升序排序）及出现次数。

输入样例1

failure is probably the fortification in your pole

it is like a peek your wallet as the thief when you

are thinking how to spend several hard-won lepta

when you are wondering whether new money it has laid

background because of you then at the heart of the

most lax alert and most low awareness and left it

godsend failed

!!!!!

输出样例1

46

the=4

it=3

you=3

and=2

are=2

is=2

most=2

of=2

when=2

your=2

输入样例2

Failure is probably The fortification in your pole!

It is like a peek your wallet as the thief when You

are thinking how to. spend several hard-won lepta.

when yoU are? wondering whether new money it has laid

background Because of: yOu?, then at the heart of the

Tom say: Who is the best? No one dare to say yes.

most lax alert and! most low awareness and\* left it

godsend failed

!!!!!

输出样例2

54

the=5

is=3

it=3

you=3

and=2

are=2

most=2

of=2

say=2

to=2

sum = 0

vd = dict()

while(1):

line = input()

if line == '!!!!!':

break

line = line.replace('!',' ').replace('.',' ').replace(',',' ').replace(':',' ').replace('\*',' ').replace('?',' ').lower()

line = ' '.join(line.split())

line = line.split()

for i in line:

if i not in vd.keys():

vd[i] = 1

sum += 1

else:

vd[i]+=1

print(sum)

aps = sorted(vd.items(),key=lambda d:d[1],reverse = True)

low=0

high=1

val=aps[0][1]

sc=0

k=1

while(sc < 10):

while(val == aps[k][1]):

high+=1

k+=1

aps[low:high]=sorted(aps[low:high],key=lambda d:d[0])

sc=sc+high-low

low = high

high+=1

val = aps[low][1]

k+=1

for i in range(10):

print('{}={}'.format(aps[i][0],aps[i][1]))

**R7-3 字典-计算字典与列表元素乘积之和 (10 分)**

先用赋值语句建立字典d = {'a':1,'b':2,'c':3,'d':4}，然后再用input()语句输入用两个英文逗号分开的3个数字并赋值给x，再将x变成列表b，计算字典d中每个元素的值与列表b中每个数字依次相乘积之总和并输出。输入输出格式见样例。

### 输入格式:

输入用两个英文逗号分开的3个数字。

### 输出格式:

输出格式见样例。

### 输入样例1:

1,2,3

### 输出样例1:

60

### 输入样例2:

5,6,7

### 输出样例2:

180

d = {'a':1,'b':2,'c':3,'d':4}

x = input().split(",")

b = list(x)

y = []

for i in d.values():

for j in b:

y += [i\*int(j)]

print(sum(y))