**2.12：**

S1 = 0

S2 = 1

S3 = 0

假设：

S1 = 1

S2 = 1

S3 = 1

**2.13：**

D = B + C + A +ABC

E = BC + B + C

**2.14：**

L = BC

L = C+AE

L = AB++B

**2.15：**

print('A',end = ' ')

print('B',end = ' ')

print('C',end = ' ')

print('D')

def conver\_1(x):#十进制转二进制

if x<2:

return([x])

r=x%2

return(conver\_1(x//2)+[r])

for i in range(0,8):

x = conver\_1(i)

while len(x)<3:

x = [0] + x

A = x[0]

B = x[1]

C = x[2]

k = (A and B and C) or (A and not B and not C) or (not A and B and not C) or (not A and not B and C)

if k == True:

k = 1

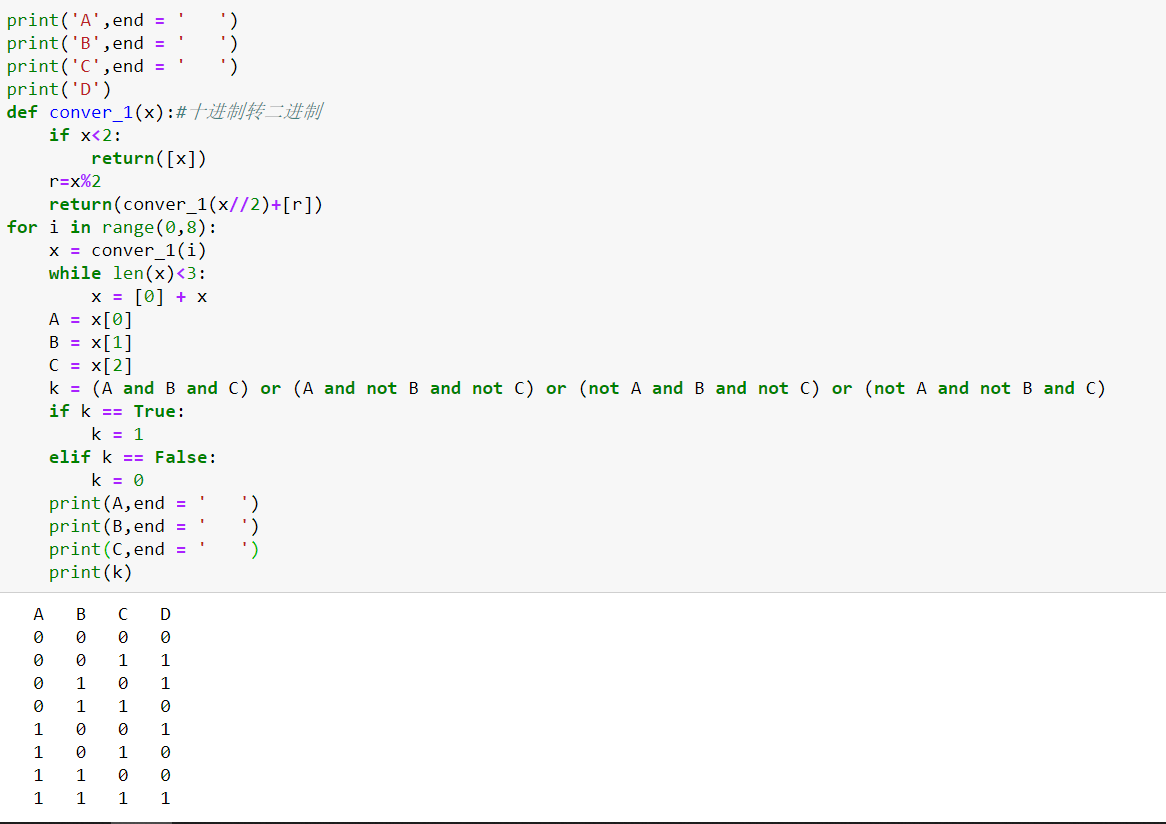
elif k == False:

k = 0

print(A,end = ' ')

print(B,end = ' ')

print(C,end = ' ')

print(k)

**2.17：**

在计算过程中，本应carry与列表相加，但其中一个列表为空时直接返回了另一个。没有进一步运算，返回值不很正确。

**2.19：**

X = 1

或X = 6

**7:**

1 01111100 00000000000000000000000

0 10001001 11100000000000000000000

**8：**

def f2(x):

d = ''

x1 = list(x)

for i2 in range(len(x1)): #取反码

if x1[i2] == '0':

x1[i2]= '1'

elif x1[i2] == '1':

x1[i2]= '0'

for i in x1:

d = d + i

d1 = int(g(d))+1

return d1

def g(x): #二进制转十进制

d = 0

weight = 2\*\*(len(x)-1)

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

if x[i] == '1':

d = d + weight

weight = weight // 2

return d

def conver\_1(x): # 十转二进制

if x<2:

return([x])

r=x%2

return(conver\_1(x//2)+[r])

def f1(x,y):

s = ''

x2 = abs(x)

x1 = conver\_1(x2)

k = []

while len(x1) < y: #补零

x1 = [0]+x1

if x >= 0:

for i in x1:

s += str(i)

return s

else:

for i1 in range(len(x1)):

k.append(x1[i1])

else:

for i2 in range(len(k)): #取反码

if k[i2] == 0:

k[i2]= 1

else:

k[i2]= 0

flag = True

for i3 in range(len(k)-1,-1,-1): #加一运算

if flag == True and k[i3] == 0:

k[i3] = 1

flag = False

elif flag == True and k[i3] == 1:

k[i3] = 0

for j in k:

s += str(j)

return s

def add(a,b): #使用字符串补位等长加法

if len(a) < len(b):

b,a = a,b

a = a[::-1] #倒序输出字符串

b = b[::-1]

while len(a) != len(b): #补位

b = b + "0"

extra = 0 #用于进位运算

new = ""

for i, j in enumerate(a):

b\_sum = int(b[i])

new = new + str((int(j) + b\_sum + extra) % 2) #加法

if int(j) + b\_sum + extra > 1: #是否进位

extra = 1

else:

extra = 0

if extra == 1: #最高位是否进位

new = new + '1'

return new[::-1]

def Mult(a,b,num):

L = ''

S1 = ''

S2 = ''

if (a > 0 and b > 0) or(a < 0 and b < 0):

a1 = conver\_1(abs(a))

b1 = conver\_1(abs(b))

for i1 in a1:

S1=S1+str(i1)

for j1 in b1:

S2=S2+str(j1)

if len(a1)<len(b1):

S1,S2 = S2,S1

for i in range(len(S2)):

if S2[len(S2)-i-1] == '1':

l = S1 + '0'\*i

L = add(L, l)

elif (a < 0 and b > 0) or (a > 0 and b < 0):

if a > 0 and b < 0:

a,b = b,a

a1 = f1(a,num)

b1 = conver\_1(b)

for j1 in b1:

S2=S2+str(j1)

for i in range(len(S2)):

if S2[len(S2)-i-1] == '1':

l = a1 + '0'\*i

L = add(L,l)

while len(L) < 12:

L = '0' + L

d = L[::-1]

d1 = d[0:12]

if a > 0 and b > 0 and d[11] == '1':

k = '溢出'

elif a < 0 and b < 0 and d[11] == '1':

k = '溢出'

elif (a < 0 and b > 0 and d[11] == '0') or (a>0 and b<0 and d[11] == '0'):

k = '溢出'

elif a == 0 and b == 0:

k = '0'

else:

d2 = d1[::-1]

if d1[11] == '1':

k0 = f2(d2)

k = '-'+str(k0)

else:

k = g(d2)

return k

a = int(input('请输入一个乘数：'))

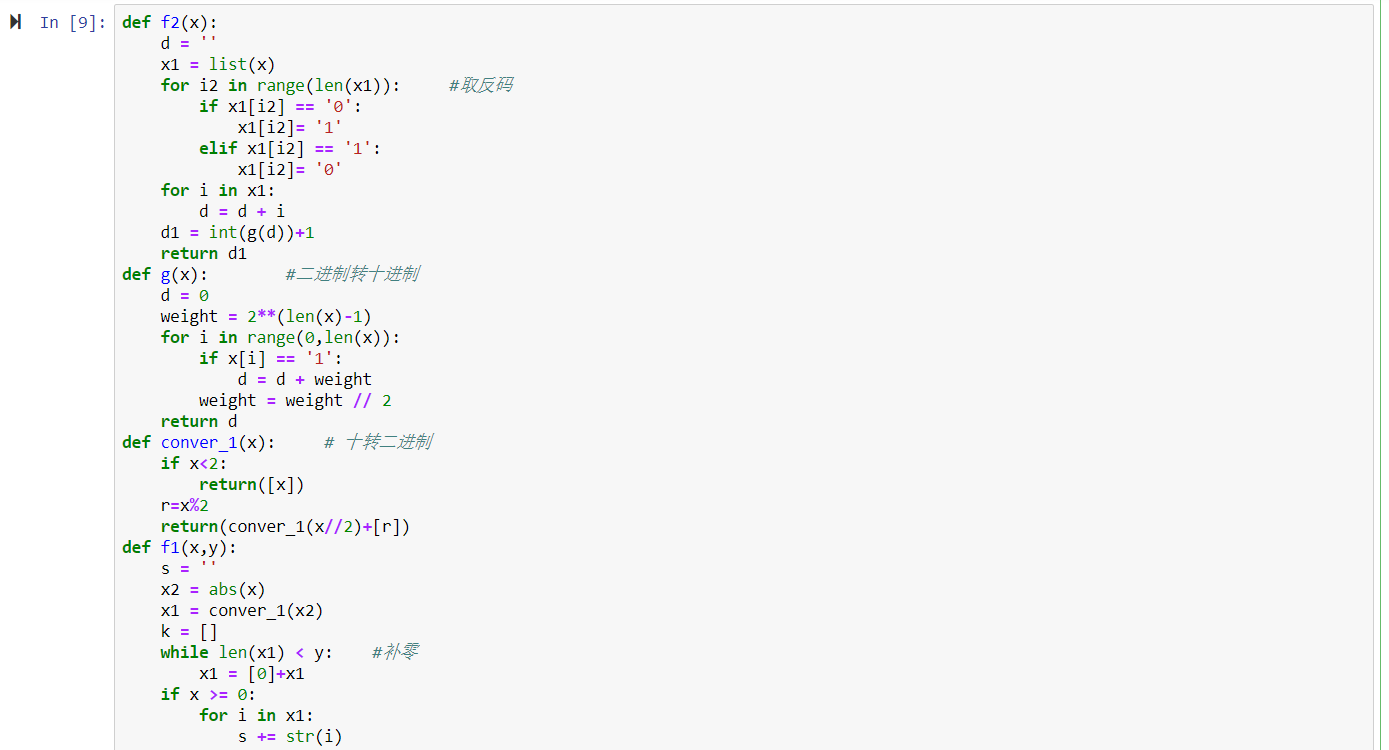
b = int(input('请输入一个乘数：'))

if (not (-2048) <= a and a < 2047) or (not (-2048) <= b and b < 2047) :

print('error')

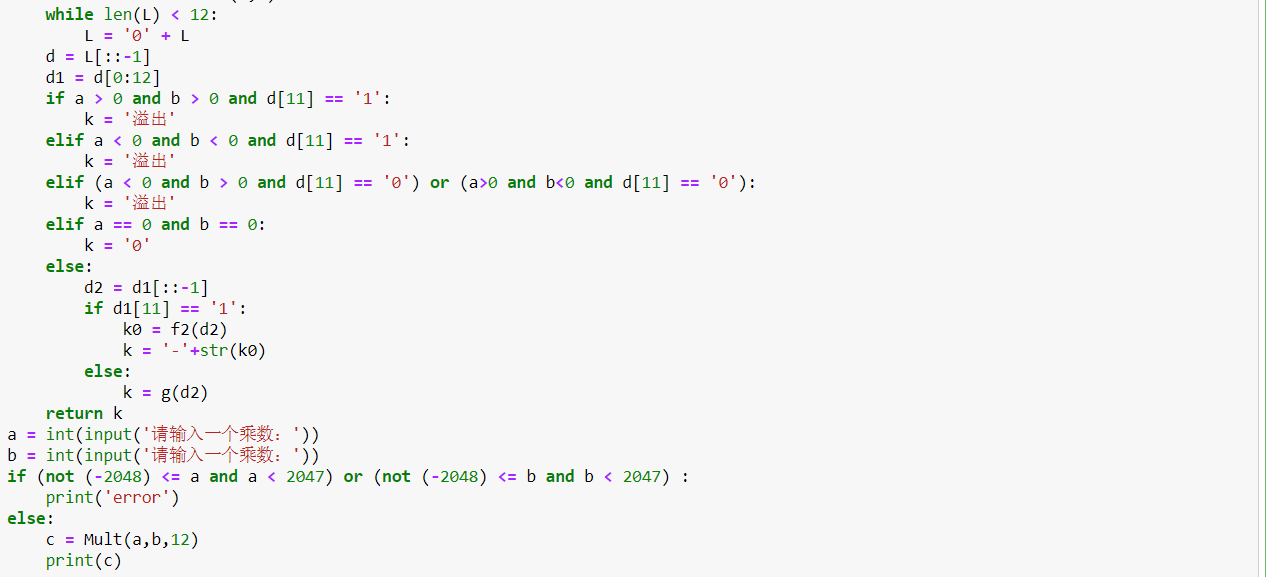
else:

c = Mult(a,b,12)

print(c)

















**9.**

**10**

**（A）（32位）**

def g0(x): # 二进制小数部分转十进制

d = 0

k = 2

for i in range(len(x)):

if x[i] == '1':

d += k\*\*(-i-1)

return d

def g(x): #二进制转十进制

d = 0

weight = 2\*\*(len(x)-1)

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

if x[i] == '1':

d = d + weight

weight = weight // 2

return d

def f1(x): #整数部分的转换

if x<2:

return([x])

r = x % 2

return(f1(x//2)+[r])

def f2(x):

k = ''

while x != 0:

k += str(int(2\*x))

x = 2\*x - int(2\*x)

return k

def f3(x):

L = f1(int(x))

if (x-int(x)) == 0:

k = '0'

else:

k = f2(x-int(x))

d = ''

for i in L:

d += str(i)

d = d+'.'+k

return d

def f4(x):

if x < 0:

k = '1'

return k

else:

k = '0'

return k

def f5(x):

m = f3(abs(x))

if m[0] == '0':

for i in range(len(m)):

if m[i] != '0' and m[i] != '.':

k = -(i - 1)

break

else:

for i in range(len(m)):

if m[i] == '.':

k = i-1

if k < -126:

s1 = '00000000'

else:

s1 = ''

l = f1(k + 127)

for j in l:

s1 = s1 + str(j)

return s1

def f6(x):

m = f3(abs(x))

if m[0] == '0':

for i in range(len(m)):

if m[i] != '0' and m[i] != '.':

k = -(i - 1)

break

else:

for i in range(len(m)):

if m[i] == '.':

k = i-1

break

if m[0] == '0':

s2 = m[abs(k) + 2:]

else:

m0 = m[1:]

s2 = m0.replace('.','')

return s2

def dec2float(x):

m = f3(abs(x))

if m[0] == '0':

for i in range(len(m)):

if m[i] != '0' and m[i] != '.':

k = -(i - 1)

break

else:

for i in range(len(m)):

if m[i] == '.':

k = i-1

break

if k > 127:

print('溢出')

elif k < -126:

s = '00000000000000000000000000000000'

else:

if x == 0:

s = ''

for i in range(32):

s = '0' + s

return s

else:

s1 = f4(x)

s2 = f5(x)

s3 = f6(x)

while len(s2) <= 7:

s2 = '0'+s2

if len(s3) > 23:

s3 = s3[0:23]

else:

while len(s3) <= 22:

s3 = s3 + '0'

s = s1+s2 +s3

return s

def float2dec(x):

if x == '0'\*32:

return '0'

else:

k = 1

if x[0] == '1':

k = -1

k0 = x[1:9]

k5 = g(k0)

k0 = int(k5)-127# 指数

k1 = x[9:]

k2 = '1'

if k0 > 0:

while k0 != 0 and k1 != '':

k2 = k2 + k1[0]

k1 = k1[1:]

k0 -= 1

k1 = g0(k1)

k4 = g(k2)

t = (int(k4) + k1)\*2\*\*k0

elif k0 < 0:

while k0 != -1:

k2 = '0' + k2

k0 += 1

k2 = k2 + k1

k2 = g0(k2)

t = k2

else:

k3 = g0(k1)

t = 1 + k3

return t \* k

x = float(input('请输入一个数：'))

if x == 0:

result = '0‘\*32

print(result)

c = float2dec(result)

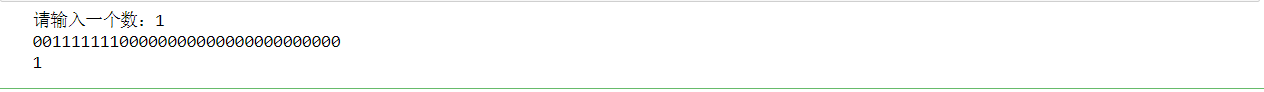
print(c)

else:

result = dec2float(x)

print(result)

c = float2dec(result)

print(c)







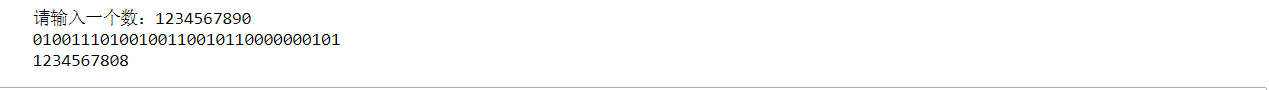
















**（B）（64位）**

def g0(x): # 二进制小数部分转十进制

d = 0

k = 2

for i in range(len(x)):

if x[i] == '1':

d += k\*\*(-i-1)

return d

def g(x): #二进制转十进制

d = 0

weight = 2\*\*(len(x)-1)

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

if x[i] == '1':

d = d + weight

weight = weight // 2

return d

def f1(x): #整数部分的转换

if x<2:

return([x])

r = x % 2

return(f1(x//2)+[r])

def f2(x):

k = ''

while x != 0:

k += str(int(2\*x))

x = 2\*x - int(2\*x)

return k

def f3(x):

L = f1(int(x))

if (x-int(x)) == 0:

k = '0'

else:

k = f2(x-int(x))

d = ''

for i in L:

d += str(i)

d = d+'.'+k

return d

def f4(x):

if x < 0:

k = '1'

return k

else:

k = '0'

return k

def f5(x):

m = f3(abs(x))

if m[0] == '0':

for i in range(len(m)):

if m[i] != '0' and m[i] != '.':

k = -(i - 1)

break

else:

for i in range(len(m)):

if m[i] == '.':

k = i-1

if k < -1023:

s1 = '00000000'

else:

s1 = ''

l = f1(k + 1023)

for j in l:

s1 = s1 + str(j)

return s1

def f6(x):

m = f3(abs(x))

if m[0] == '0':

for i in range(len(m)):

if m[i] != '0' and m[i] != '.':

k = -(i - 1)

break

else:

for i in range(len(m)):

if m[i] == '.':

k = i-1

break

if m[0] == '0':

s2 = m[abs(k) + 2:]

else:

m0 = m[1:]

s2 = m0.replace('.','')

return s2

def dec2float(x):

m = f3(abs(x))

if m[0] == '0':

for i in range(len(m)):

if m[i] != '0' and m[i] != '.':

k = -(i - 1)

break

else:

for i in range(len(m)):

if m[i] == '.':

k = i-1

break

if k > 1024:

print('溢出')

elif k < -1023:

s = '0'\*64

else:

if x == 0:

s = ''

for i in range(64):

s = '0' + s

return s

else:

s1 = f4(x)

s2 = f5(x)

s3 = f6(x)

while len(s2) <= 10:

s2 = '0'+s2

if len(s3) > 52:

s3 = s3[0:52]

else:

while len(s3) <= 51:

s3 = s3 + '0'

s = s1+s2 +s3

return s

def float2dec(x):

if x == '0'\*64:

return '0'

else:

k = 1

if x[0] == '1':

k = -1

k0 = x[1:12]

k5 = g(k0)

k0 = int(k5)-1023# 指数

k1 = x[12:]

k2 = '1'

if k0 > 0:

while k0 != 0 and k1 != '':

k2 = k2 + k1[0]

k1 = k1[1:]

k0 -= 1

k1 = g0(k1)

k4 = g(k2)

t = (int(k4) + k1)\*2\*\*k0

elif k0 < 0:

while k0 != -1:

k2 = '0' + k2

k0 += 1

k2 = k2 + k1

k2 = g0(k2)

t = k2

else:

k3 = g0(k1)

t = 1 + k3

return t \* k

x = float(input('请输入一个数：'))

if x == 0:

result = '0'\*64

print(result)

c = float2dec(result)

print(c)

else:

result = dec2float(x)

print(result)

c = float2dec(result)

print(c)

