**专题一：进制转换专题**

**一、基础**

1. **十进制转二进制**
2. **二进制转十进制**

**二、提升**

1. **十六进制转十进制**

将5位以内的十六进制正整数转化成十进制数的python程序，功能如下：输入一个十六进制数，输出十进制数。可实现上述功能的python程序如下，请在划线处填入合适的代码。

x=input("")

i=1

result=0

flag=True

while i<=len(x) and flag==True:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_#①

if ch>="0" and ch<="9":

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ #②

elif ch>="A" and ch<="F":

result=result\*16+(ord(ch)-ord("A")+10)

elif ch>="a" and ch<="f":

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_#③

else:

flag=False

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_#④

if flag==True:

print(result)

**2.二进制加法**

编写“二进制加法”程序。程序运行时，分两行输入两个二进制数，输出两个二进制相加后的结果。

输入：

1101

110

输出：10011

x = input().strip() #strip()函数是去除行末的空格与回车符

y = input().strip()

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_#①

if dis < 0:

x, y = y, x

dis = -dis

for i in range(dis):

y = "0" + y

jw = 0

ans = ""

for i in range(len(x) - 1, -1, -1):

a = int(x[i]) + int(y[i]) + jw

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_#②

ans = str(a % 2) + ans

if\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ :#③

ans = "1" + ans

print(ans)

**3.“5421BCD”码**

“5421BCD”码是一种采用四位二进制数表示一位十进制数的编码，其各位的权依次位5、4、2、1，并要求大于等于5的十进制数所对应的编码最高位为1（如十进制数7所对应的“5421BCD”码为1010）。转换时，将十进制数从右至左按位转换成对应的“5421BCD”码，然后依次连接。

输入：一个正整数，如：579

输出：一行5421BCD码，如：100010101100

x = int(input())

ans = ""

while x > 0:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_#①

a1 = a // 5

a = a % 5

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_#②

a3 = a // 2

a = a % 2

x = x // 10

ans =\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_#③

print(ans)

**4.求一个数的反码**

在计算机中负数的反码计算如下：给出十进制数x（-127<=x<0），将x的绝对值转化为七位二进制数（不足7位时高位补0），然后逐位取反，最左边增加符号位1得到八位二进制数反码。例如十进制数-7，将7转换为七位二进制数为0000111，逐位取反变成1111000，最左边增加符号位1 得到八位二进制数反码为11111000。编写一个反码转化程序，输入十进制负数x（-127<=x<0），输出对应的反码。

输入：一个整数x（-127<=x<=0）

输出：该数的反码

x = int(input())

ans = ""

c = 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_#①

while x > 0:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_#②

ans = str(r) + ans

x = x // 2

c = c + 1

for i in\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_：#③

ans = "1" + ans

print( ans)

**5.求一个数的补码**

在计算机系统中，数值一律用补码来表示和存储。求负整数的补码是将该数对应的正数转换为二进制数（不足 7 位时高位补 0），最高位作为符号位为 1，其余每位二进制数取反（ 1 变成 0,0 变成 1），末位加 1。编写一个补码转化程序，输入十进制负数x（-127<=x<0），输出对应的补码。

输入：-12

输出：补码11110100

x = abs(int(input("输入十进制数:")))

a = [0] \* 7

for i in range(7):

a[i] = 1 - x % 2

x = x // 2

i = 0

a[0] = a[0] + 1

while\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_:#①

a[i] = 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_#②

i = i + 1

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_#③

for i in range(7):

ans = ans + str(a[6 - i])

print(ans)