2023 年 GESP9 月认证 Python 四级试卷解析

CCF 编程能力等级认证，英文名 Grade Examination of Software Programming（以下简称 GESP），由中国计算机学会发起并主办，是为青少年计算机和编程学习者提供学业能力验证的平台。GESP 覆盖中小学全学段，符合条件的青少年均可参加认证。GESP 旨在提升青少年计算机和编程教育水平，推广和普及青少年计算机和编程教育。

GESP 考察语言为图形化（Scratch）编程、Python 编程及 C++编程，主要考察学生掌握相关编程知识和操作能力，熟悉编程各项基础知识和理论框架，通过设定不同等级的考试目标，让学生具备编程从简单的程序到复杂程序设计的编程能力，为后期专业化编程学习打下良好基础。

本次为大家带来的是 2023 年 9 月份 Python 四级认证真题解析。

一、单选题（每题 2 分，共 30 分）

1.人们所使用的手机上安装的 App 通常指的是（ ）。

A. 一款操作系统

B. 一款应用软件

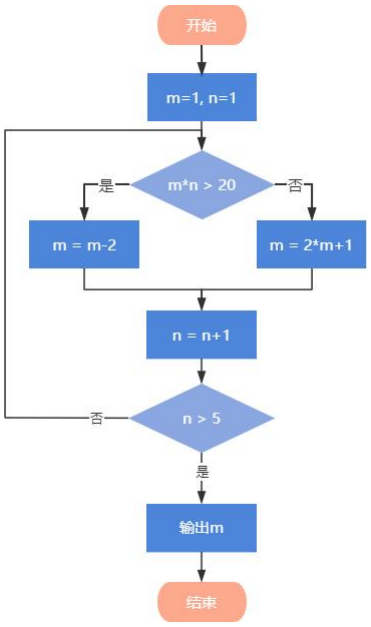
C. 一种通话设备

D. 以上都不对

【答案】B

【解析】本题属于考察计算机基础知识。App 是 Application 的缩写，通常指的是应用程序。在计算机领域，App 是指一种可以在特定操作系统上运行的软件程序，它为用户提供了特定的功能和服务。例如，手机上的微信、支付宝等应用就是 App。故正确答案为 B 选项。

2.下列流程图的输出结果是（）？



A. 9

B. 7

C. 5

D. 11

【答案】A

【解析】本题属于考察流程图的知识。该流程图主要进行条件判断，当 m*n>20 时，执行 m=m-2，n=n+1。否则，执行 m=2*m+1，n=n+1。再判断 n>5，满足条件输出 m，不满足条件时，回到第一个判断条件。该流程图判断 5 次，最后 m=9。故正确答案为 A 选项。

3.下面有关 print() 函数的说法，错误的是（）。

A. print()函数的 sep 和 end 参数为带有默认值的命名关键字参数。

B. print()函数可以输出多个表达式的值，其参数为变长参数。

C.如果 print()函数同时使用 sep 和 end 参数，则要求 sep 在前 end 在后。

D. print()函数可以输出多个不同数据类型表达式的值。

【答案】C

【解析】本题属于考察 Python 内置函数 print()函数参数的知识。在 Python 的 print( )函数中，sep 和 end 参数用于控制输出的分隔符和结束符。sep 参数用于指

定分隔符，默认为一个空格；end 参数用于指定结束符，默认为换行符。如果同时使用 sep 和 end 参数，那么它们的顺序并不影响输出的结果，即 sep 在前或 end 在前都可以实现预期的效果。故正确答案为 C 选项。

4.下面 Python 代码执行后输出是（）。

```
listA = [1,2,2,2,4,4,4]
print(listA)
for i in listA:
    if i % 2 == 0:
        listA.remove(i)
print(listA)
```

A. [1]

B. [1, 2, 4]

C. [1, 2, 4, 4]

D.触发异常

【答案】C

【解析】本题属于考察 Python 内置函数 remove()函数调用的知识。按照执行顺序，符合判断条件的元素被删除之后，后面的元素会前移，再次调用时会判断列表的下一个元素。本题中，要求移除能整除 2 的数据，第一次执行完 remove()列表为[1,2,2,4,4,4]，第二次执行完列表为[1,2,4,4,4]，第三次执行完为[1,2,4,4]。故正确答案为 C 选项。

5.下面代码执行后输出是（）。

```
def Fx(a,b):
    a += 10
    b = 11
    c = 22
    return a + b + c

a, b, c = 1, 2, 3
Fx(a, b)
print(a, b, c)
```

A. 1, 2, 3

B. 1, 2, 22

C. 11, 11, 22

D. 11, 11, 3

【答案】A

【解析】本题属于考察 Python 函数参数、作用域的知识。在 Python 中，函数是一段具有特定功能的可重用代码。当我们调用一个函数时，需要传递一些值给函数，这些值被称为参数。函数的参数可以分为形参和实参两种。此题输出语句位于函数体外，输出的是全局变量的 a，b，c。输出的是 1，2，3，故正确答案为 B 选项。

6.下面代码执行后输出是（）。

```
def pushNum(lst,num):
    lst.append(num)
    return lst

lstA = [1, 2, 3]
pushNum(lstA, 4)
print(lstA)
```

A. [1, 2, 3]

B. [1, 2, 3, 4]

C. [4]

D. None

【答案】B

【解析】本题属于考察 Python 函数参数、作用域的知识。无论函数传递的参数的可变还是不可变，只要针对参数使用赋值语句，会在函数内部修改局部变量的引用，不会影响到外部变量的引用，但是传递的参数是可变类型（列表（List）、字典（Dictionary）和集合（Set）），在函数内部使用方法修改了数据的内容，同样会影响到外部的数据。本题中，当函数接收到列表参数时，它实际上是接收到了对该列表对象的引用。因此，在函数内部对列表进行的任何修改都会影响到

外部的列表对象。故正确答案为 B 选项。

7.下面 Python 代码执行后输出是（ ）。

```
def mergeData(tplData,num):
    tplData = tplData + (num,)
    return tplData

tpl = (1, 2, 3)
print(mergeData(tpl, 4), end = ",")
print(tpl)
```

A. (1, 2, 3),(1, 2, 3)

B. (1, 2, 3, 4),(1, 2, 3)

C. (1, 2, 3),(1, 2, 3, 4)

D. (1, 2, 3, 4),(1, 2, 3, 4)

【答案】B

【解析】本题属于考察 Python 作用域和元组基本知识。元组（tuple）是 Python 中的一种不可变序列类型，它由多个元素组成，元素之间用逗号分隔。在函数内修改其数据将形成新的数据，不影响函数外。故正确答案为 B 选项。

8.下面 Python 代码执行后输出是（ ）。

```
def bubbleSort(lst):
    n = len(lst)
    for i in range(n):
        for j in range(n-i-1):
            if lst[j] > lst[j+1]:
                lst[j], lst[j+1] =
lst[j+1], lst[j]

lstData = [11, 2, 3, 7, 15]
bubbleSort(lstData)
print(lstData)
```

A. [2, 3, 7, 11, 15]

B. [15, 11, 7, 3, 2]

C. [11, 2, 3, 7, 15]

D. None

【答案】A

【解析】本题属于考察冒泡排序算法的知识。冒泡排序是一种简单的排序算法。它重复地遍历要排序的数列，一次比较两个元素，如果他们的顺序不符合要求就把他们交换过来。重复进行直到没有再需要交换，也就是说该数据已经排序完成。这个算法的名字由来是因为越小的元素会经由交换慢慢“浮”到数列的顶端。故正确答案为 B 选项。

9.上题 bubbleSort() 函数的时间复杂度是（ ）。

A. O(n)

B. $O(n^2)$

C. $O(n \log n)$

D. O(1)

【答案】B

【解析】本题属于考察冒泡算法复杂度的估算知识。如果数据是正序，有 n 个数，只需要走一趟即可完成排序。所需的比较次数 C 和记录移动次数 M 均达到最小值，即：Cmin=n-1;Mmin=0;所以，冒泡排序最好的时间复杂度为 O(n)。如果数据是反序的，则需要进行 n-1 趟排序。每趟排序要进行 n-i 次比较(1≤i≤n-1)，且每次比较都必须移动记录三次来达到交换记录位置。在这种情况下，比较和移动次数均达到最大值 $O(n^2)$。故正确答案为 B 选项。

10.下面 Python 代码中的 dictA 存储为字典，key（键）为 i 和 j 的组合，value（值）为 i*j，形如{(1,1):1, (1,2):2}，横线处应填上代码是（ ）。

```
dict99 = {}
for i in range(1,9+1):
    for j in range(i,9+1):
        _________________ = i * j
print(dict99)
```

A. dict99[(i, j)]

B. dict99[[i, j]]

C. dict99(i, j)

D. dict99{i, j}

【答案】A

【解析】本题属于考察 Python 复合数据类型的嵌套知识。题目要求形如{(1,1):1, (1,2):2}，按照字典的赋值格式 dict[key] = value，故正确答案为 A 选项。

11.下面 Python 代码中的 dictA 变量存储形如 {1: [1], 2: [1, 2], 3: [1, 3], 4: [1, 2, 4], 5: [1, 5], 6: [1, 2, 3, 6]}的数据，即 1~99 之间每个整数的因数（所有能被整除的正整数），横线处应填入是（）。

```
dictA = {}
for i in range(1,100):
    dictA[i] =
_____________________________
print(dictA)
```

A. [j for j in range(i) if i % j != 0]

B. [j for j in range(1,i+1) if i % j == 0]

C. [j for j in range(1,i+1) if i % j]

D. [j for j in range(i) if i % j == 0]

【答案】B

【解析】本题属于考察 Python 复合数据类型的嵌套和列表推导式的知识。每个键对应的值都是能被 i 整除的正整数，包括 i。所以应使用 range(1,i+1)，同时判断是否能被 i 整除，即余数是否为 0。故正确答案为 B 选项。

12.要打开一个已经存在的图片文件并读取但不修改数据，则打开模式应该设定为（ ）。

A. wb

B. w+

C. rb

D. r+

【答案】C

【解析】本题属于考察 Python 文件读写操作知识。在 Python 中，open（）函数用于打开文件并返回一个文件对象。open（）函数的第二个参数是文件模式，它决定了文件的访问方式。要打开一个已经存在的图片文件并读取但不修改数据，则打开模式应该设定为 rb，rb 以二进制格式打开一个文件用于只读。文件指针将会放在文件的开头。wb 是以二进制格式打开一个文件只用于写入，w+打开一个文件用于读写，如果该文件已存在则打开文件，并从开头开始编辑，即原有内容会被删除。如果该文件不存在，创建新文件。r+以读写方式打开文件，如果文件不存在则报错。如果文件存在，可以从头开始读取和写入。故正确答案为 C 选项。

13.下列 Python 代码执行时如果输入 3.14 ，将输出的是（ ）。

```
try:
    m, n =
map(int,input().split(","))
    print(m,n)
    res = m/n
except ZeroDivisionError:
    print(1, end = "#")
except:
    print(2, end = "#")
else:
    print(3, end = "#")
finally:
    print(4, end = "#")
```

A. 2#

B. 1#4#

C. 2#4#

D. 2#3#4#

【答案】C

【解析】本题属于考察 Python 异常处理和内置函数结合使用的知识。map()接收一个函数和一个 list，并把函数依次作用在 list 的每个元素上，得到一个新的 list 并返回。split()将一个字符串按照指定的分隔符进行分割，返回一个由分割后的子字符串组成的列表。input()函数返回的是字符串类型。异常处理机制 try...except...else...finally 是一种用于捕获和处理程序运行过程中可能出现的异常。它包含以下几个部分：try：尝试执行一段可能引发异常的代码。如果这段代码没有引发异常，那么 except 和 else 部分将被跳过，程序将继续执行后面的代码。except：当 try 中的代码引发异常时，程序将跳转到 except 部分执行。你可以在 except 后面指定要捕获的异常类型，例如，except ZeroDivisionError 异常表示捕获除数为零的异常。else：当 try 中的代码没有引发异常时，程序将执行 else 部分的代码。注意，一个 try 语句块中只能有一个 else 部分。finally：无论 try 中的代码是否引发异常，程序都会执行 finally 部分的代码。故正确答案为 B 选项。

14.以下选项在 Python 中能输出 (1, 4, 9, 16, 25, 36, 49, 64, 81, 100) 的是（ ）。

A. print(tuple(i**2 for i in range(10)))

B. print(tuple([i*i for i in range(1,10+1)]))

C. print(tuple(i*i for i in range(10+1)))

D. print(tuple(map(lambda x:x**2,range(10))))

【答案】B

【解析】本题属于考察 Python 复合数据类型的嵌套知识。(1, 4, 9, 16, 25, 36, 49, 64, 81, 100)分别是 1-10 的二次方，即 i*i 或 i**2。所以 range()函数的起始值为 1，结束值为 10+1。A，C，D 选项的 range（）函数取值范围不对，故正确答案为 B 选项。

15.Python 赋值语句是 lstA = [6, 7, 8, 9] ，删除值为 8 的元素，错误的语句( )。

A. lstA.remove(8)

B. lstA.pop(2)

C. del　lstA[2]

D. lstA.del[2]

【答案】D

【解析】本题属于考察 Python 内置函数的调用和列表知识。在 Python 中，列表是一种有序的集合，可以随时添加和删除其中的元素。删除列表元素的方法可以使用 del 关键字删除指定索引的元素，格式为 del 列表名[索引值]；2.使用 remove()方法删除列表中的指定值的第一个匹配项；3.使用 pop()方法删除并返回指定索引的元素，默认为最后一个元素；4.使用 clear()方法清空列表中的所有元素。故正确答案为 D 选项。

二、判断题（每题 2 分，共 20 分）

1. 我们常说的互联网（Internet）是一个覆盖全球的广域网络，它不属于任何一个国家。

【答案】正确√

【解析】本题考察计算机网络的基本知识。互联网，又称国际网络，指的是网络与网络之间所串连成的庞大网络，这些网络以一组通用的协议相连，形成逻辑上的单一巨大国际网络。互联网是全球性的。这就意味着这个网络不管是谁发明了它，是属于全人类的。故说法正确。

2. 我国第一台大型通用电子计算机使用的逻辑部件是晶体管。

【答案】错误×

【解析】本题考察计算机网络的基本知识。1958 年 8 月 1 日，中国科学院计算技术研究所和北京有线电厂(国营 738 厂)根据苏联提供的 M-3 小型机技术资料制成的“八一”型通用电子管计算机(又称 103 机)完成了四条指令的运行，标志着中国人制造的第一架通用数字电子计算机正式诞生。时隔一年多，1959 年 9 月，根据苏联有关计算机技术资料制成的 104 大型通用电子计算机通过试运算，运

算速度提升到每秒 1 万次。《人民日报》为此发表消息，正式宣告中国第一台大型通用电子计算机试制成功。机体内由半导体锗二极管和电子管组成。故说法错误。

3. Python 的内置函数 sorted() 函数是稳定排序。

【答案】正确√

【解析】本题考察 sorted（）函数相关知识和算法稳定概念，sorted()函数的实现基于 Timsort 算法，Timsort 算法在排序时，先将待排序的序列分割成若干个子序列，对每个子序列进行排序，然后将已排序的子序列合并成一个新的有序序列，直到整个序列排好序为止。是结合了合并排序和插入排序而得出的排序算法。如果一个排序确保不会改变比较结果相等的元素的相对顺序就称其为稳定的。故说法正确。

4. 对包含 N 个元素列表（list）进行冒泡排序算法，其时间复杂度是$O(N^2)$。

【答案】正确√

【解析】本题属于考察冒泡算法复杂度的估算知识。如果数据是正序，有 n 个数，只需要走一趟即可完成排序。所需的比较次数 C 和记录移动次数 M 均达到最小值，即：Cmin=n-1;Mmin=0;所以，冒泡排序最好的时间复杂度为 O(n)。如果数据是反序的，则需要进行 n-1 趟排序。每趟排序要进行 n-i 次比较(1≤i≤n-1)，且每次比较都必须移动记录三次来达到交换记录位置。在这种情况下，比较和移动次数均达到最大值 O(n^2)。故说法正确。

5. ()+() 在 Python 中是合法的表达式，其值为 () 。

【答案】正确√

【解析】本题考察元组和运算符使用的知识。在 Python 中，通过“+”将两个或多个元组合并成一个新的元组是拼接元组。元组是一种不可变的序列类型，用于存储一组有序的数据。通过拼接元组，我们可以将多个元组合并在一起，形成一个更大的元组。故说法正确。

6. 下面代码中的 Nums.txt 文本文件中含有 0-9 共计 10 个数字，分为两行存储，第 1 行为 0-4，第 2 行为 5-9，程序执行后将输出 10。

```
rFile = open("Nums.txt","r")
print(len(rFile.read()))
```

【答案】错误×

【解析】本题考察文件读取的知识，read()用于从文件中读取内容。当使用 open()打开一个文件时，会返回一个文件对象，然后可以使用该对象的 read()方法来读取文件内容。会将文件中的所有字符读取到一个字符串中，并返回这个字符串。结果应为 11,故说法错误。

7. Python 文本文件读取函数 readlines() 能按行读取文本文件，且返回值为 list 类型。

【答案】正确√

【解析】本题考察文件读取的知识，readlines()是 Python 中文件对象的方法，用于一次性读取文件中的所有行，并返回一个包含所有行的列表。每一行都是一个字符串。故说法正确。

8. 下面的 Python 代码执行后最后一行将输出没有偶数的 lst。

```
lst = list(range(10))
for i in lst:
    if i % 2 == 0:
        del i
print(lst)
```

【答案】错误×

【解析】本题属于考察 Python 复合数据类型的嵌套知识。del 是 Python 中的一个关键字，用于删除对象。它可以删除变量、列表元素、字典元素等。在循环中删除列表元素时，需要使用索引来定位要删除的元素，而不是直接使用变量名。

因此，应该使用 del l[i]而不是 del i。所以最后输出为[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]。故说法错误。

9．在与异常处理相关的关键字中，finally 所属内容不管是否发生异常都将会被执行。

【答案】正确√

【解析】本题属于考察 Python 异常处理。finally 无论 try 中的代码是否引发异常，程序都会执行 finally 部分的代码，故说法正确。

10. global 关键字只能用于自定义函数内，其功能是允许在函数内修改全局变量的值。

【答案】正确√

【解析】本题属于考察 Python 作用域的知识。global 关键字只适用于在函数内部修改全局变量的值，如果需要在函数内部创建新的全局变量，不需要使用 global 关键字。故说法正确。

三、编程题（每题 25 分，共 50 分）

1. 进制转换

【问题描述】

N 进制数指的是逢 N 进一的计数制。例如，人们日常生活中大多使用十进制计数，而计算机底层则一般使用二进制。除此之外，八进制和十六进制在一些场合也是常用的计数制（十六进制中，一般使用字母 A 至 F 表示十至十五）。

在本题中，我们将给出 N 个不同进制的数。你需要分别把它们转换成十进制数。

【提示】

对于任意一个 L 位 K 进制数，假设其最右边的数位为第 0 位，最左边的数位为第 L-1 位，我们只需要将其第 i 位的数码乘以权值K^i，再将每位的结果相加，即可得到原 K 进制数对应的十进制数。下面是两个例子：

1. 八进制数 1362 对应的十进制数为$1 \times 8^3 + 3 \times 8^2 + 6 \times 8^1 + 2 \times 8^0 = 754$；
2. 十六进制数 3F0 对应的十进制数为$3 \times 16^2 + 15 \times 16^1 + 0 \times 16^0 = 1008$。

【输入描述】

输入的第一行为一个十进制表示的整数 N。接下来 N 行，每行一个整数 K，随后是一个空格，紧接着是一个 K 进制数，表示需要转换的数。保证所有 K 进制数均由数字和大写字母组成，且不以 0 开头。保证 K 进制数合法。

保证$N \leq 1000$；保证$2 \leq K \leq 16$,保证所有 K 进制数的位数不超过 9。

【输出描述】

输出 N 行，每一个十进制数，表示对应 K 进制数的十进制数值。

【样例输入 1】

2

8　1362

16　3F0

【样例输出 1】

754

1008

【样例输入 2】

2

2　11011

10　123456789

【样例输出 2】

27

12345679

【题目大意】

首先，输入要转换的行数，用 N 表示。接下来输入 N 行，每行格式为：K　K 进制数。将给出 N 个不同进制的数分别把它们转换成十进制数输出。

【解题思路】

本题主要考察内置函数，嵌套循环和进制转换的使用。

1. 首先获取一个整数 n。

2. 进行 n 次循环，在每次循环中，将该行输入分割成两个部分 k 和 st。k 设置为一个整数，st 是一个字符串。

3. 将 st 逆序，遍历 st 中的每个字符，使用 enumerate()函数将 st 组合为一个索引序列，同时列出数据 ch 和数据下标 j。使用 if 语句根据字符的类型（数字或字母）计算出其对应的权重，并将这些权重乘以 k 的相应次方 j，最后将所有的乘积相加得到结果 ans。

【参考程序】

```
n = int(input())
for i in range(n):
    k, st = input().strip().split(' ')
    k = int(k)
    ans = 0
    for j, ch in enumerate(reversed(st)):
        if ch <= '9':
            bit = ord(ch) - ord('0')
        else:
            bit = ord(ch) - ord('A') + 10
        ans += bit * (k ** j)
    print(ans)
```

2. 变长编码

【问题描述】

小明刚刚学习了三种整数编码方式：原码、反码、补码，并了解到计算机存储整数通常使用补码。但他总是觉得，生活中很少用到2^{31}-1 这么大的数，生活中常用的 0~100 这种数也同样需要用 4 个字节的补码表示，太浪费了些。热爱学习的小明通过搜索，发现了一种正整数的变长编码方式。这种编码方式的规则如下：

1. 对于给定的正整数，首先将其表达为二进制形式。例如，$(0)_{10} = (0)_{2}$，$(926)_{10} = (1110011110)_{2}$ 。

2. 将二进制数从低位到高位切分成每组 7bit，不足 7bit 的在高位用 0 填补。例如，$(0)_{2}$变为 00000000 一组，$(1110011110)_{2}$变为 0011110 和 0000111 的两组。

3. 由代表低位的组开始，为其加入最高位。如果这组是最后一组，则在最高位填上 0，否则在最高位填上 1。于是，0 的变长编码为 00000000 一个字节，926 的变长编码为 10011110 和 00000111 两个字节。

这种编码方式可以用更少的字节表达比较小的数，也可以用很多的字节表达非常大的数。例如，987654321012345678 的二进制为 $(0001101\ 1011010\ 0110110\ 1001011\ 1110100\ 0100110\ 1001000\ 0010110\ 1001110)_{2}$，于是它的变长编码为（十六进制表示）CE 96 C8 A6 F4 CB B6 DA 0D，共 9 个字节。你能通过编写程序，找到一个正整数的变长编码吗？

【输入描述】

输入第一行，包含一个正整数 N。约定$0 \leq N \leq 10^{18}$。

【输出描述】

输出一行，输出 N 对应的变长编码的每个字节，每个字节均以 2 位十六进制表示（其中，A-F 使用大写字母表示），两个字节间以空格分隔。

【样例输入 1】

0

【样例输出 1】

00

【样例输入 2】

926

【样例输出 2】

9E 07

【样例输入 3】

987654321012345678

【样例输出 3】

CE 96 C8 A6 F4 CB B6 DA 0D

【题目大意】

输入一个正整数 N，转换为二进制数，将二进制数从低位到高位切分成每组 7bit，不足 7bit 的在高位用 0 填补。由代表低位的组开始，为其加入最高位。如果这组是最后一组，则在最高位填上 0，否则在最高位填上 1。最后转换为十六进制数输出。

【解题思路】

本题主要考察内置函数，for…else..语句和字符串方法的使用。

1. 输入正整数 N，使用 bin()函数，转为二进制数并去掉二进制数的前缀，再赋值给 N。

2. 判断二进制 N 是否是 7 位，不是择在前面补 0。

3．设置一个空列表 bList，用来存放二进制数，使用 for 循环遍历 N，每 7 位位一个列表项。

4．逆序存储在 bList 中，除了最后一个列表项，“1”+每个列表项。最后一项，“0”+每个列表项。

5．设置一个空字符串 rst，遍历 bList，使用 int()函数将每个遍历元素（即列表项）转为 10 进制数，再使用 hex()函数转为 16 进制数并使用字符串切片去除进制前缀“0x”。使用 upper()函数将字母转为大写字母，使用 zfill()函数在字符串的左侧填充零，直到达到指定的长度 2。最后用空字符串分割。

6.去掉最后的空字符输出 rst。

【参考程序】

```
N = int(input())
N = bin(N)[2:]
if len(N) % 7 != 0:
    N = "0" * (7 - len(N) % 7) + N
bList = []
for i in range(0, len(N), 7):
    bList.append(N[i:i + 7])
bList = bList[::-1]
for i, b7 in enumerate(bList[:-1]):
    bList[i] = "1" + b7
else:
```

```
    bList[-1] = "0" + bList[-1]
rst = ""
for b8 in bList:
    rst += hex(int(b8, 2))[2:].upper().zfill(2) + " "
rst = rst[:-1]
print(rst)
```