

## 2023 年 3 月 GESP Python 二级试卷解析

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 3 月份，Python 二级考试试卷真题解析。

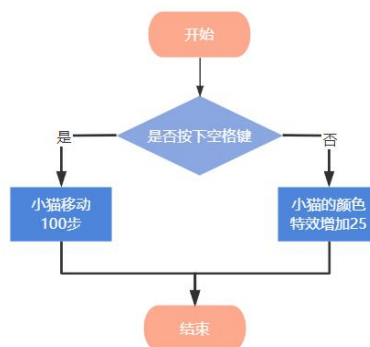
1. 以下存储器中的数据不会受到附近强磁场干扰的是（ ）。

- A. 硬盘
- B. U 盘
- C. 内存
- D. 光盘

**【答案】D**

**【解析】**本题属于考察计算机的存储于网络相关知识；存储设备中，硬盘、U 潘和内存都借助于将电信号转换成二进制数字信号读写数据，当强磁场出现时，会对信息读写造成干扰；而光盘利用激光对数据进行读写，属于物理结构，不受磁场干扰，故本题选 D。

2. 下列流程图，属于计算机的哪种程序结构？（ ）。



- A. 顺序结构
- B. 循环结构
- C. 分支结构
- D. 数据结构

【答案】C

【解析】本题属于考察 Python 中的选择结构(也叫分支结构)：通过设置条件并判断条件是否成立，给程序设置不同的结果，带来程序的分支，这种结构称作分支结构，故正确答案为 C 选项。

3. 以下选项中，不符合 Python 语言变量命名规则的是( )。

- A. Print
- B. print
- C. int
- D. for

【答案】D

【解析】本题考察的是 Python 中变量名的命名规则：变量名中能且只能包括数字、字母、下划线，且不能以数字开头，同时注意不能够使用已有的关键字作为变量名，故选 D。需注意的是，选项中 B、C 属于 Python 中的内置函数，定义变量名时不会导致程序报错，但会引发函数功能无法正确使用的问题，同样也不建议使用内置函数作为变量名。

4. 为使下程序代码正常执行，横线处应填写的代码是( )。

```
#随机函数的使用
_____
print(random.random())
```

- A. import random
- B. import Random
- C. from random import random
- D. import random as Random

【答案】A

【解析】本题考察 Python 中标准库 random 随机库的使用：使用随机功能的函数前，需要导入 random 库；B 选项库名拼写错误；C 选项导入方式只导入了 random 函数，没有导入 random 库；D 选项库名拼写和导入方式都错误；正确答案为 A 选项。

5. 下面有关 Python 循环中有关 break 语句的说法，正确的是( )。

- A. 在多层循环（循环嵌套）中，有且只能有一个 break 语句；
- B. 在多层循环（循环嵌套）中，内外层循环最终执行一次 break 语句；

- C. 在多层循环（循环嵌套）中，当内层 break 语句被执行时，将中断循环并跳到最外层循环之后；
- D. 在多层循环（循环嵌套）中，当内层 break 语句被执行时，只能中断所在层的循环；

【答案】D

【解析】本题属于考察 Python 中的循环结构与 break 关键字的用法；在多层循环中，可以在不同层存在多个 break 语句，并且每个 break 关键字被执行时，只能终止当前所在层的循环，故本题选择 D 选项。

6. 在 Python 中，表达式 `ord("a") - ord("A")` 的值为（ ）。

- A. 32
- B. -32
- C. 26
- D. 错误

【答案】A

【解析】本题属于考察编程中 ASCII 码的相关知识；首先每个单个字符，按照 ASCII 码的编码规则都可以用一个整数表示，内置函数 `ord()` 就可以求出某个单字符的 ASCII 码值；小写字母 a 码值为 97，大写字母 A 码值为 65，故本题选 A。另外，还需了解小写字母的 ASCII 值大于大写字母，大写字母和小写字母之间还有其他字符，因此其值为正整数，且不为 26。

7. 在 Python 中，`print("123" * 2)` 的值为（ ）。

- A. 123
- B. 246
- C. 123123
- D. 错误

【答案】C

【解析】本题主要考察 Python 中不同数据类型进行运算时的规则；Python 中定义字符串型\*整型，代表将字符串重复 N 次，如 `"a"*2` 为 `"aa"`，`"1"*3` 为 `"111"`，所以本题正确答案为 C 选项。

8. 执行以下 Python 程序代码后，输出结果是（ ）。

```
for i in range(10):
    for j in range(i):
        break
    else:
        print(i)
```

- A. 没有输出
- B. 11

C. 10

D. 9

【答案】D

【解析】本题主要考察 for...else... 循环结构的用法；当使用 for...else... 结构时，如果循环没有被 break 终止，会在循环结束后执行 else 中的语句；程序中 break 终止的只是内层循环，所以当外层循环结束后，会执行 else 中的语句；由于 range(10)，变量 i 的取值范围为 0~9，最后 i 的值为 9，故此题选 D。

9. 下列代码用于判断年份是否为闰年，请在横线处填上相应代码。闰年的判断规则是能被 400 整除或者能被 4 整除但不能被 100 整除。（ ）。

```
#闰年判断
N = int(input("请输入年份: "))

if _____:
    print(f"{N}年是闰年")
else:
    print(f"{N}年不是闰年")
```

A.  $N \% 400 == 0$  or  $N \% 4 == 0$  and  $N \% 100 != 0$

B.  $N \% 400 == 0$  and  $N \% 4 == 0$  and  $N \% 100 != 0$

C.  $N \% 400 == 0$  or  $N \% 4 == 0$  or  $N \% 100 != 0$

D.  $N \% 400 == 0$  and  $N \% 4 == 0$  or  $N \% 100 != 0$

【答案】A

【解析】本题主要考察运算符中求余运算、逻辑运算的使用；判断闰年的条件是：1. 年份能被 4 整除，但不能被 100 整除；2. 年份能被 400 整除，以上条件满足一个即可；条件 1 和条件 2 是“或”的逻辑关系，条件 1 中的两个条件是“与”的逻辑关系，选项中 A 满足条件逻辑，故正确答案为 A。

10. 执行以下 Python 程序代码后，输出结果是（ ）。

```
cnt = 0 #保存循环次数
for i in range(1, 10):
    for j in range(i, 10):
        if i * j % 2 == 0:
            break
        cnt += 1

print(cnt)
```

A. 5

B. 9

C. 81

D. 100

【答案】A

【解析】本题考察了 for 循环中 range() 函数参数与变量 i 的关系，与循环结构中流程控制关键字 break 的使用；根据给出的循环嵌套结构，可得知外层循环变量 i 的范围为从 1 到 9，内层循环变量 j 的范围为从 i 到 9；当外层循环 i 取偶数时， $i*j\%2==0$  条件成立，执行 break 终止内层循环，故 cnt 变量不会增加；所以当 i 取值为 1、3、5、6、9 时，cnt 变量会增加 1，共 5 次，故本题选 A。

11. 执行以下 Python 程序代码后，输出结果是（ ）。

```
cnt = 0 #保存循环次数
for i in range(1, 10):
    for j in range(i, 10):
        if i * j % 2 == 0:
            break
    cnt += 1

print(cnt)
```

A. 5

B. 9

C. 81

D. 100

【答案】B

【解析】本题考察了 for 循环中 range() 函数参数与变量 i 的关系，与循环结构中流程控制关键字 break 的使用；可分析代码得知变量 i 的取值从 1 到 9，程序运行 9 次；由于 cnt+=1 的代码放置在内层循环外，所以每次内层循环结束后 cnt 变量都增加 1，跟内层条件无关，cnt 变量共增加 9 次，故本题选 B 选项。

12. 执行下面 Python 代码并输入 1，其输出是（ ）。

```
N=int(input("请输入整数: "))

divCount = 0 #保存次数

for i in range(1,N+1):
    if N % i == 0:
        divCount +=1

if divCount == 2:
    print(f"{N}是质数,整除次数为{divCount}")
else:
    print(f"{N}不是质数,整除次数为{divCount}")
```

- A. 1 不是质数, 整除次数为 1
- B. 1 不是质数, 整除次数为 2
- C. 1 是质数, 整除次数为 1
- D. 1 是质数, 整除次数为 2

【答案】A

【解析】本题主要考察了 for 循环中变量 i 的取值范围和条件判断结构；当输入 1 时，N 值为 1，变量 i 的取值也为 1；所以条件  $N\%i==0$  相当于  $1\%1==0$ ，条件成立，divCount 变量增加 1；结束循环后判断条件  $divCount==2$  不成立，所以执行 else 中的语句，故正确答案为 A 选项。

13. 在以下 Python 代码中，横线应填入（ ）。

```
#判断N是否对称数，如:1221
N=int(input("请输入正整数: "))

i = N
res = 0 #保存结果

while i != 0:
    res _____
    i = i // 10

if res == N:
    print(f"{N}是对称数")
else:
    print(f"{N}不是对称数")
```

- A.  $= res * 10 + i \% 10$
- B.  $*= 10 + i \% 10$
- C.  $+= i \% 10$
- D.  $*= i \% 10$

【答案】A

【解析】本题主要考察了 while 循环的使用和取整取余的应用；对称数的主要思路是：将原数字从后向前求出每位数，并且把每位数倒序排列，最终弄比较倒序数字和原数字是否相同，得到是否是对称数； $res*10$  表示将上次结果扩大 10 倍， $i\%10$  表示求出上次数字的个位数， $i=i//10$  表示去除上次结果的个位数；使用 while 循环重复直到 i 为 0 时，判断条件即可；故本题目答案为 A 选项。

14. 在下面 Python 代码横线处应填上的代码是（ ）。

```

#斐波那契数列形如: 1,1,2,3,5,8,13,.....
#从第3项开始为前面两项之和, 第1项编号为1

N = int(input("请输入序号: "))

prev1 = 1 #前1值
prev2 = 1 #前2值

for i in range(3, N+1):
    _____

print(prev1)

```

- A. prev2 = prev1
- B. prev1, prev2 = prev2, prev1
- C. prev2 = prev1; prev1 = prev1 + prev2
- D. prev2, prev1 = prev1, prev1+prev2

【答案】D

【解析】本题主要考察变量赋值的特点；根据斐波那契数列定义，每项的数值等于前两项的和，所以需要将 prev1 和 prev2 相加存入变量，C 选项中分步操作已经修改了 prev2 的值，写法错误，故本题正确答案为 D 选项。

15. 输出如下形式的九九乘法表，横线处应填上的代码是（ ）。

```

1*1= 1
1*2= 2 2*2= 4
1*3= 3 2*3= 6 3*3= 9
1*4= 4 2*4= 8 3*4=12 4*4=16
1*5= 5 2*5=10 3*5=15 4*5=20 5*5=25
1*6= 6 2*6=12 3*6=18 4*6=24 5*6=30 6*6=36
1*7= 7 2*7=14 3*7=21 4*7=28 5*7=35 6*7=42 7*7=49
1*8= 8 2*8=16 3*8=24 4*8=32 5*8=40 6*8=48 7*8=56 8*8=64
1*9= 9 2*9=18 3*9=27 4*9=36 5*9=45 6*9=54 7*9=63 8*9=72 9*9=81

```

```

#九九乘法表
for i in range(1,10):
    for j in range(1,i+1):
        if _____:
            print(f"{j}*{i}= {i*j} ",end="")
        else:
            print(f"{j}*{i}={i*j} ",end="")
    print()

```

- A.  $i * j > 10$
- B.  $i * j < 10$
- C.  $i * j != 10$
- D.  $i * j == 10$

【答案】B

【解析】本题主要考察的是对程序细节的观察；程序中给出的是双层的循环嵌套结构，在内层循环中设置 if...else... 结构，通过观察发现，两条输出语句的差别仅有 {i\*j} 前是否有空格，再结合输出的乘法表，发现结果是个位数的数字前有一个空格，所以条件应该是  $i*j < 10$ ，此题正确答案是 B 选项。

## 二、判断题 (每题 2 分, 共 20 分)

1. 明明和笑笑在“小庙会”上分别抽到一个 4GB 和 4096MB 的 U 盘，容量大的盘是笑笑的 ( )。

【答案】错误 ×

【解析】本题考察了计算机的存储相关的知识；在计算机中，GB 和 MB 的换算关系为  $1\text{GB} = 1024\text{MB}$ ，所以 4GB 和 4096MB 的容量相同，故此题错误。

2. IPv4 的地址通常用“点分十进制”的表示形式，形如 (a.b.c.d)，其中 a、b、c、d 都是  $1 \sim 255$  之间的十进制整数 ( )。

【答案】错误 ×

【解析】本题主要考察计算机网络的相关知识；对于 IPv4 地址的描述中，a、b、c、d 的取值范围应该是从  $0 \sim 255$  之间的十进制整数，故此题错误。

3. 当执行 Python 表达式 `int(input())` 输入 3.14 后，其结果为 3。 ( )

【答案】错误 ×

【解析】本题考察了 python 中数据类型转换的相关知识；`int()` 函数可以将只包含整数的字符串，转换为十进制整数，但当 `input()` 函数接收到的字符串为 3.14 时，无法进行数据类型转换，程序会抛出异常，故此题错误。

4. 假设已正确执行 `import random`，Python 表达式 `int(random.random())` 的值一定为 0。 ( )

【答案】正确 ✓

【解析】本题考察了 python 中数据类型转换的相关知识；`int()` 函数可以将浮点数类型数据转换成整数，转换规则为截去小数点后的数值，只保留整数；`random()` 函数会生成 0 到 1 之间的小数，不包括 1，故结果一定为 0，此题正确。

5. Python 表达式 `round(299792.458, -3)` 的值为 300000.0。 ( )

【答案】正确 ✓

【解析】本题考察了 python 中数据类型转换的相关知识；`int()` 函数可以将浮点数类型数据转换成整数，转换规则为截去小数点后的数值，只保留整数；`random()` 函数会生成 0 到 1 之间的小数，不包括 1，故结果一定为 0，此题正确。

6. 在 Python 语言中，不仅 if 语句可以有 else 子句，while 和 for 循环也都可以有 else 子句。 ( )



【答案】正确 ✓

【解析】本题主要考察 Python 中的循环结构；for 循环和 while 循环可以在循环结束后搭配 else 关键字使用，表示当循环中没有执行 break 语句而终止循环后，执行 else 中的语句，故此题正确。

7. Python 表达式 `bool("") == bool(" ")` 的值为 True，其中  表示空格。（ ）

【答案】错误 ✕

【解析】本题主要考察 Python 中数据类型的相关知识；bool 类型中只有 True（真）、False（假）两个值，程序中定义将非 0、非空（空字符串、空列表等）的值视为 False，其他值为 True；`bool("") == bool(" ")` 式子中，左侧为 False，右侧为 True，== 运算符两侧的值不相等，故此题错误。

8. 在 Python 语言中，`print()` 因为没有参数而错误。（ ）

【答案】错误 ✕

【解析】本题考察了 `print()` 输出语句的基本用法；`print()` 函数可以不设置参数，效果为输出一个空行，故此题错误。

9. 在 Python 语句 `print(4, 5, sep = "", end = ">")` 中，sep 和 end 不可以交换位置。（ ）

【答案】错误 ✕

【解析】本题考察了 `print()` 输出语句中的参数设置；`print()` 输出函数的必要参数，是若干个需要输出的值；其中的 sep、end 参数，分别用来输出的设置间隔、结尾部分的内容，使用指定参数名的传参方式，可以调换参数的顺序，故此题错误。

10. 先执行 Python 代码 `a, b = 5, 6`，然后执行语句 `print(a or b)` 将输出 True。（ ）

【答案】错误 ✕

【解析】本题考察了逻辑运算符 or 的用法；逻辑运算符 or，当参与运算的两个值都为数字时，如果两个数字都为 0，则运算结果返回 0，否则从左向右返回第一个遇到的非 0 数字；本题中 `a or b` 相当于 `5 or 6`，运算结果返回 5，此题错误。

### 三、编程题 (每题 25 分, 共 50 分)

#### 1. 字母阵列

##### 【问题描述】

输入一个正整数 N, 输出 N 行 N 列字母, 字母为大写英文字母, 执行效果如下。输入正整数 27, 输出 27 行字母, 每行 27 个字母。第一行从 A 开始, 第二行 B 开始。当到达 Z, 再从 A 开始。

```
27
ABCDEFGH IJKLMNOPQRSTUVWXYZA
BCDEFGH IJKLMNOPQRSTUVWXYZAB
CDEFGH IJKLMNOPQRSTUVWXYZABC
DEFGH IJKLMNOPQRSTUVWXYZABCD
EFGH IJKLMNOPQRSTUVWXYZABCDE
FGH IJKLMNOPQRSTUVWXYZABCDEF
GH IJKLMNOPQRSTUVWXYZABCDEFGH
HIJKLMNOPQRSTUVWXYZABCDEFGH I
JKLMNOPQRSTUVWXYZABCDEFGH J
KLMNOPQRSTUVWXYZABCDEFGH JK
LMNOPQRSTUVWXYZABCDEFGH JKL
MNOPQRSTUVWXYZABCDEFGH JKLM
NOPQRSTUVWXYZABCDEFGH JKLMN
OPQRSTUVWXYZABCDEFGH JKLMNO
PQRSTUVWXYZABCDEFGH JKLMNOP
QRSTUVWXYZABCDEFGH JKLMNOPQ
RSTUVWXYZABCDEFGH JKLMNOPQR
STUVWXYZABCDEFGH JKLMNOPQRS
TUVWXYZABCDEFGH JKLMNOPQRST
UVWXYZABCDEFGH JKLMNOPQRSTU
VWXYZABCDEFGH JKLMNOPQRSTUV
WXYZABCDEFGH JKLMNOPQRSTUVW
XYZABCDEFGH JKLMNOPQRSTUVWX
YZABCDEFGH JKLMNOPQRSTUVWXY
ZABCDEFGH JKLMNOPQRSTUVWXYZ
ABCDEFGH IJKLMNOPQRSTUVWXYZA
```

##### 【输入描述】

输入一行, 一个大于等于 1 的正整数 N。

##### 【输出描述】

输出上图所示的字母阵列。

##### 【样例输入 1】

5

##### 【样例输出 1】

```
ABCDE
BCDEF
CDEFG
DEFGH
EFGHI
```

##### 【样例输入 2】

10

### 【样例输出 2】

```
ABCDEFGHIJ
BCDEFGHIJK
CDEFGHIJKL
DEFGHIJKLM
EFGHIJKLMN
FGHIJKLMNO
GHIJKLMNOP
HIJKLMNOPQ
IJKLMNOPQR
JKLMNOPQRS
```

**【题目大意】**根据输入的数字，输入整数 N，输出 N 行大写字母，每行字母数量为 N 个，内容为按字母表顺序依次排列的字母；第一行和第一列皆从 A 开始，向后排列 N 个字符(超过 26 个时从 A 从头排列输出)。

### 【解题思路】

1. 观察出规律后，先列出双层循环嵌套，输出 N 行，每行 N 个字符；
2. 借助 chr() 和 ord() 函数，处理对应的 ASCII 值或字符，最终输出指定字符；
3. 外层循环中的变量 i，控制内层循环每行的起始字母(第一列的起始字母)；
4. 内层循环中的变量 j，控制字母从 i 开始不断向后增加；
5. 用 cnt 变量记录需要输出的 ASCII 码值，并考虑当 cnt 大于 26 的处理方法。

### 【参考程序】

```
N = int(input())    #输入整数 N
for i in range(N):
    cnt = i          #cnt 变量用来记录每行第一个字母的 ASCII 码值(j 的初始值)
    for j in range(N):
        if cnt > 25:      #当 cnt 大于 25 时，表示输出数量超过 26
            cnt = cnt % 26  #对 26 取余，从 A 开始继续输出字母
        print(chr(ord("A") + cnt), end = "")  #处理字符并输出
        cnt += 1          #ASCII 码增加 1
    print()             #内层循环结束后输出空行
```

## 2. 角谷猜想

### 【问题描述】

角谷猜想是指任何一个正整数如果是偶数则变为其一半,如果是奇数则变为其 3 倍加 1, 最终都将变为 1。如果为 1, 则按照规则将变为 4, 4 则变为 2, 2 则变为 1, 进入无限循环。该猜想尚未得到数学证明;

本题先后输入两个正整数, 输出该范围内角谷猜想变化步骤最多的数及其步骤; 如果输入的第 1 个数大于第 2 个数, 则交换, 如 20 10, 则计算出 10 到 20 (含) 之间变化步骤最多的数及其步骤。

### 【输入描述】

输入两行, 第 1 行输入范围起点, 回车后第 2 行输入范围的终点。

**特别提示:** 常规程序中, 输入时好习惯是有提示。考试时由于系统限定, 输入时所有 `input()` 函数不可有提示信息。

### 【输出描述】

输出两行, 先输出变化步骤最大的数, 然后输出步骤数量。

步骤计算时, 包含数本身, 例如: 10 的变化过程是 10, 5, 16, 8, 4, 2, 1, 变化步骤数为 7。

### 【样例输入 1】

10

20

### 【样例输出 1】

18

21

### 【样例输入 2】

100

200

### 【样例输出 2】

171

125

### 【样例输入 3】

500

300

### 【样例输出 2】

327

144

**【题目大意】**根据角谷猜想的规则，根据输入的两个数字，模拟并计算两个数字范围内，最终推算为 1 总共需要的推算次数最多的数字和推算次数。

**【解题思路】**

1. 首先需要确认，应使用 while 循环根据角谷猜想规则去计算推算次数；
2. 接下来想到可以在外部设置 for 循环，嵌套 while 循环计算两个输入数范围内每个数字的推算次数；
3. 输入并保存两个数字 a、b，同时需考虑并处理两个数字的大小关系(保证  $a < b$ )；
4. 创建变量记录最多推算次数，和最多推算次数的数字；
5. 设置循环嵌套结构，运算、处理数据，并按要求输出最终结果。

**【参考程序】**

#指定范围内角谷猜想最多步骤

First = int(input())       #输入两个数字

Second = int(input())

if First > Second:       #比较大小，确保小的数字在前

    First, Second = Second, First

maxStep = 0               #记录最多推算次数

maxNum = First           #记录最多推算次数的数字

for i in range(First, Second+1):   #for 循环列举两数范围内的所有数字

    N = i                   #将数字存入变量 N 并计算 N 的推算次数

    stepCount = 1           #初始化数字 N 的推算次数为 1

        while N != 1:       #根据角谷猜想，当 N 等于 1 时停止

            stepCount += 1   #将推算次数增加 1

            if N % 2 == 0:   #角谷猜想处理奇、偶数的规则

                N = N // 2

            else:

                N = N \* 3 + 1

    if stepCount > maxStep:   #判断推算次数是否为最大值

        maxStep = stepCount   #记录最大推算次数

        maxNum = i           #记录最大推算次数的数字

print(maxNum)               #循环结束后按要求输出结果

print(maxStep)