

GESP Python 二级试卷

(满分：100 分 考试时间：90 分钟)

学校：_____

姓名：_____

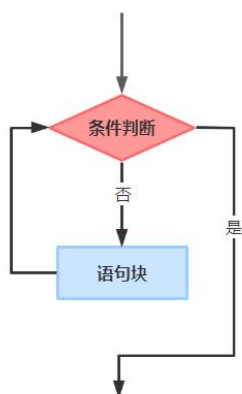
题目	一	二	三	总分
得分				

一、单选题 (每题 2 分, 共 30 分)

题号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
答案	D	D	D	C	A	D	A	B	C	D	C	A	A	A	C

1、 高级语言编写的程序需要经过以下（ ）操作，可以生成在计算机上运行的可执行代码。

- A. 编辑
- B. 保存
- C. 调试
- D. 编译



2、 能够实现下面流程图功能的伪代码是（ ）。

- A. if 条件判断 then 语句块
- B. if 条件判断 then 什么也不做 else 语句块

- C. while 条件判断 do 语句块
- D. while not 条件判断 do 语句块
- 3、在 Python 代码中，变量 a 的值为 4，则表达式 $5 < a < 20$ 的值是（ ）。
- A. true
- B. false
- C. True
- D. False
- 4、Python 表达式 `round(8/3, 3)` 的值是（ ）。
- A. 2
- B. 3
- C. 2.667
- D. 2.666

```
import math
print(math.floor(-5.67))
```

- 5、下面 Python 代码执行后输出是（ ）。
- A. -6
- B. -5
- C. 0
- D. 报错

```
a, b = 3, 2
if a > b:
    a = b
    b = a
print(a, b, sep=",")
```

- 6、下面 Python 代码执行后的输出是（ ）。
- A. 3,2

- B. 2,3
- C. 3,3
- D. 2,2

```
m, n = 13, 3
z = 13 % 5
if z < 5:
    print(5)
elif z < 4:
    print(4)
elif z < 3:
    print(3)
else:
    print("0-2")
```

7、下面 Python 代码执行后的输出是（ ）。

- A. 5
- B. 4
- C. 3
- D. 0-2

```
cnt = 0
for i in range(5):
    for j in range(i):
        cnt += 1
print(cnt)
```

8、下面 Python 代码执行后的输出是（ ）。

- A. 15
- B. 10
- C. 5

D. 0

9、求 m 到 n 之间的间隔数数量， m 可能大于 n ，如 3 到 6 之间间隔数数量为 2,6 和 3 之

```
#求m-n之间的间隔数量，如3到6之间间隔数量是2
m = int(input("请输入起点数:"))
n = int(input("请输入终点数:"))

gap = 1 #设定间隔值
if m > n :
    _____

cnt = 0
for i in range(m+1, n, gap):
    cnt += 1
print(cnt)
```

间的间隔数同样是 2。根据上述题意，在 Python 代码横线处应填上（ ）。

- A. gap = 1
- B. gap = 0
- C. gap = -1
- D. gap = gap

10、正整数 N 除以 16 的余数为 0-15，将 0-15 对应为不同字符，0-9 仍然为 0-9,10-15 依次对应为大写应为字母 A-F。根据上述题意，在 Python 代码横线处应填上（ ）。

- A. chr(ord("A")+Remainder-10)
- B. chr(ord("A")+Remainder)
- C. chr(Remainder)
- D. chr(ord("A")+Remainder-10)

11、下面 Python 代码执行后的输出是（ ）。

- A. 55
- B. 18

C. 9

```
cnt = 0
for i in range(10):
    for i in range(i):
        if i % 2:
            break
        cnt += 1
print(cnt)
```

D. 33

12、 有关下面 Python 代码执行的说法，正确的是（ ）。

- A. 代码执行后如果输入 5，则将输出 5
- B. 代码执行后如果输入 5，则将再输入 5 次，输出由先后输入值决定
- C. 代码输入执行后如果不输入小于 0 的数，将无限循环
- D. 代码执行将报错。

13、 两个正整数的最大公约数是指能被两个数都整除且最大，如 12 和 18 能被 2、3 和 6 整除，但 6 最大，所以 12 和 18 的最大公约数是 6。两个质数如 13 和 17 的最大公约

```
m = int(input())
n = int(input())
if m > n :
    m, n = n, m
for i in _____:
    if m % i == 0 and n % i == 0:
        print(f"最大公约数={i}")
        break
```

数是 1。根据上述题意，在 Python 代码横线处应填上（ ）。

- A. range(m, 0, -1)
- B. range(m)
- C. range(m, 1, -1)

D. range(1, m)

14、产生 m 和 n 之间的随机整数，包括 n 和 m。根据上述题意，在 Python 代码横线

```
import random
m = int(input())
n = int(input())
if m > n:
    m, n = n, m
print(_____)
```

处应填上（ ）。

- A. m + 1 + int((n - m) * random.random())
- B. m + 1 + int((n - m) * random.random)
- C. m + 1 + int((n - m)) * random.random
- D. m + 1 + (n - m) * int(random.random())

```
for i in range(8, 2, -2):
    for j in range(i):
        print("ok")
```

15、下面 Python 代码执行后输出“OK”（ ）行。

- A. 0
- B. 12
- C. 18
- D. 20

二、判断题 (每题 2 分, 共 20 分)

题号	1	2	3	4	5	6	7	8	9	10
答案	X	X	X	X	√	X	X	X	√	X

1. 诞生于 1986 年的中华学习机 CEC-I 入选了 2021 年的 CCF 计算机历史记忆（一类），它的内存只有 64KB。当时的汉字编码字符集 GB2312 中共有 6763 个汉字，假如每个汉字用 2 个字节编码，将整个 GB2312 汉字字符集都放入 CEC-I 的内存，也只占用了不超过 1/5 的内存空间。
2. 域名是由一串用点分隔的名字来标识互联网上一个计算机或计算机组的名称，CCF 编程能力等级认证官方网站的域名是 `gesp.ccf.org.cn`，其中顶级域名是 `gesp`。
3. Python 表达式 `5.0//2` 的值为整数 2。
4. 执行 Python 代码 `a=f'{101:2.2f}'` 后，变量 `a` 的类型为 `float`，其值 101.00。
5. 当 Python 执行循环时，如果执行到 `break` 语句，这终止其所在循环。

```
n = int(input())#假设输入为负整数、零和正整数
cnt = 0
while n:
    n -= 1
    cnt += 1
print(cnt)
```

6. 执行以下 Python 代码，如果 `n` 值为负整数，将输出 0。
7. 执行以下 Python 代码，将输出 π 的值。
8. Python 是一种低级语言，适合初学者程序设计入门。
9. `{1,1}` 在 Python 中是合法的表达式。
10. 以下 Python 代码执行时将报错，因为 `a` 初始被赋值为整数，随后将其赋值为字符串，

```
a = 123
a = "ABC"
```

故错误。

```
import math
print(math.pi())
```

三、编程题

第 1 题

【问题描述】

小明刚刚学习了素数的概念：如果一个大于 1 的正整数，除了 1 和它自身外，不能被其他正整数整除，则这个正整数是素数。现在，小明想找到两个正整数 **A** 和 **B** 之间（包括 **A** 和 **B**）有多少个素数。

【输入描述】

输入 2 行，第一行包含正整数 **A**，第二行包含正整数 **B**。约定 $2 \leq A \leq B \leq 1000$ 。

【输出描述】

输出一行，包含一个整数 **C**，表示找到 **C** 个素数。

【样例输入 1】

2
10

【样例输出 1】

4

【样例解释 1】

在 2 和 10 之间有 4 个素数，分别为：2、3、5、7。

【样例输入 2】

98
100

【样例输出 2】

0

【参考程序】

```
n = int(input())
```

```
m = int(input())
```

```
tnt = 0
```

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

```
    for j in range(2,i):
```

```
        if i % j == 0:
```

```
break

else:

    tnt += 1

print(tnt)
```

第 2 题

【问题描述】

1. 判断一个正整数是否为自幂数；
2. 自幂数是指 N 位数各位数字 N 次方之和是本身，如 153 是 3 位数，其每位数的 3 次方之和是 153 本身，因此 153 是自幂数，1634 是 4 位数，其每位数的 4 次方之和是 1634 本身；
3. 首先输入一个整数 M 表示随后有 M 行每行有一个整数，输出每个整数是否为自幂数，如是则输出 T，否则 F。

【输入描述】

1. 输入正整数 M ($1 \leq M \leq 100$)。不考虑输入不合规情形如负数、负整数、零或其他字符串等；
2. 从第 2 行开始，输入 M 行每行一个正整数 ($1 \leq M \leq 2147483647$)；
3. **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】

1. 输出对应正整数是否为自幂数，如是则输出一个英文字母 T，否则一个英文字母 F；
2. **特别提示：**注意输出字母为大写，小写将判为错误，前后没有空格等。

【样例输入 1】

```
3
152
111
153
```

【样例输出 1】

```
F
F
T
```

【样例输入 2】

5

8208

548834

88593477

12345

5432

【样例输出 2】

T

T

T

F

F

【参考程序】

#自幂数

N = int(input()) #输入数据的数量

for i in range(N):

 M = int(input()) #输入判断的数

 howDigit = 0

 #获得输入整数的位数

 tmp = M

 while tmp != 0:

 howDigit += 1

 tmp //= 10 #执行完本步，M 将减少 1 位

 tnt = 0

 tmp = M

 #计算各位幂次方的和

 while tmp != 0:

```
tnt = tnt+ ( tmp % 10) ** howDigit
```

```
tmp //= 10
```

```
#判断是否为自幂数
```

```
if tnt == M:
```

```
    print("T")
```

```
else:
```

```
    print("F")
```