

GESP Python 一级样题卷

(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	C	B	B	A	B	A	B	C	D	A	C	D	C	B	A

1. 以下选项中, 不符合 Python 语言变量命名规则的是()。

- A. scoreList
- B. score_list
- C. -scoreList
- D. _scoreList

2. 下面有关 `print(a, b, sep = ", ", end = "#")` 语句的说法, 不正确的是()。

- A. `print(a, b, end = "#", sep = ", ")`与题干语句写法等效
- B. `print(end = "#", sep = ", ", a, b)`与题干语句写法等效
- C. 题干语句中 a 和 b 之间的间隔符为英文逗号
- D. 题干语句执行后最后一个字符是#

3. 下面有关 `turtle.backward(n)` 的描述, 正确的是()。

- A. n 不可以为负数或 0
- B. n 可以是负数也可以正数, 甚至可以为 0
- C. `turtle.backward(n)` 只能绘制垂直或水平直线

- D. 海龟的朝向与 n 值有关，当 n 为正数或负数时，则维持原方向，负数则与原方向相反

4. 有关下面 Python 代码，正确说法是（ ）。

```
import turtle
turtle.pensize(1)
turtle.fillcolor("red")

turtle.begin_fill()

turtle.forward(100)
turtle.left(90)
turtle.forward(100)

turtle.end_fill()
```

- A. 上述代码将形成填充颜色为红色的三角形
- B. 上述代码将形成两条垂直直线，尚不能构成三角形
- C. 上述代码因为不能构成封闭图形，故不能填充
- D. 上述代码因为 `turtle.fillcolor()` 不在 `turtle.begin_fill()` 和 `turtle.end_fill()` 之间，因此不能不能填充

5. 下面 Python 代码执行后输出如下图形，请在第 4 行 `range` 后括号内填入正确的代码（ ）。

```
import turtle
turtle.pensize(4)

for i in range( ): # 【请在括号内填入正确的内容】
    if i%2==0:
        turtle.pencolor("red")
        turtle.forward(50)
    else:
        turtle.pencolor("blue")
        turtle.forward(25)
```

- A. 4
- B. 5
- C. 1 , 5
- D. 1 , 6

6. 下面 Python 代码执行后显示的图形是 () 。

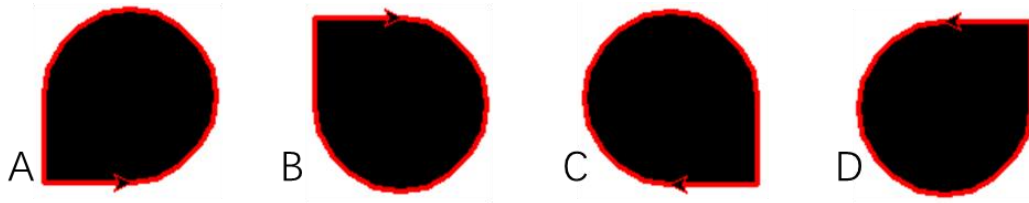
```
import turtle

turtle.pensize( 3 )
turtle.pencolor( "red" )

turtle.begin_fill( )

turtle.circle( 50 , 270 )
turtle.forward( 50 )
turtle.left( 90 )
turtle.forward( 50 )

turtle.end_fill( )
```



7. 下面 Python 代码执行后显示的图形是 ()。

```
import turtle

turtle.pensize(3)
turtle.pencolor("red")

turtle.begin_fill()

for i in range(4):
    turtle.forward(50)
    turtle.left(90)

turtle.end_fill()
```



8. 下面 Python 执行后的输出结果是 ()。

```
for i in range(1,10):
    continue
print(i,end="#")
```

- A. 1#2#3#4#5#6#7#8#9#10
- B. 1#2#3#4#5#6#7#8#9#
- C. 9#
- D. 9

9. 以下 Python 代码执行后输出结果是()。

```
loopCount=0 #存储循环次数

for i in range(10):
    loopCount += 1
    i *= 2

print(loopCount)
```

- A. 4
- B. 5
- C. 9
- D. 10

10. 下面 Python 代码的功能是统计非小写字母的数量，在【】之间应填入代码是()。

```
myStr="The Great Wall is famous in the world."

notLowerCount=0
for c in myStr:
    if 【】:
        notLowerCount += 1

print(notLowerCount)
```

- A. not ("a"<=c<="z")
- B. "a"<=c<="z"
- C. c<"a" and c>"z"
- D. not (c<"a" and c>"z")

11. 下面 Python 代码执行后输出是 ()。

```
Sum=0
for i in range(1,10,3):
    if i%2 == 0:
        Sum += i
    else:
        continue
print(Sum)
```

- A. 0
- B. 1
- C. 4
- D. 20

12. 执行以下 Python 代码后，数据结果是()。

```
x,y = 10 , 20
z = x
x = y
y = z
print( x , y , z , sep = ",")
```

- A. 10,10,20
- B. 10,20,20
- C. 20,20,10
- D. 20,10,10

13. 下面 Python 代码，执行后输出是()。

```
c = "*"

for i in range( 1 , 5 ):
    c *= i

print( len( c ) )
```

- A. 0
- B. 1
- C. 24
- D. 120

14. 某小朋友说他家台式机内存是 16G，这 16G 的单位是()。

- A. bit (比特)
- B. Byte (字节)
- C. Hz (赫兹)
- D. g (克)

15. 下面有关 Python 的说法，错误的是()。

- A. Python 是编译型高级语言，运行效率高
- B. Python 是面向对象的程序设计语言
- C. Python 有丰富的扩展库，是 Python 的重要特色
- D. Python 是跨平台解释型高级语言

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	F	F	F	F	T	T	F	F	T	F

1. Python 代码 `turtle.circle(100)` 将绘制一个以坐标原点为中心半径为 100 像素的圆。 ()
2. Python 代码 `turtle.goto(100 , 100)` 将绘制从当前坐标点到坐标 (100 , 100) 之间直线。 ()
3. 按照 Python 变量命名规则, 变量不可命名为 `input`, 因为有名称为 `input` 的函数。 ()
4. Python 表达式 `"10" ** 2` 的值为 100。 ()
5. Python 表达式 `5 * 2 ** 3 // 100` 的值为 0。 ()
6. Python 语句 `n = int(input())` 执行时如输入 3.1415926 将报错。 ()
7. Python 代码 `print("□" == "")` 【注: □表示英文空格】执行后将输出 `false`。 ()
8. Python 代码 `print("Python"[1])` 执行后将输出字母 P。 ()
9. 某手机“机身内存 1TB”其实相当于计算机的外存为 1TB。 ()
10. 编写 Python 代码时如执行“保存”操作, 相当于将代码保存到内存之中, 关机后可再次打开或执行或再次编辑。 ()

三、编程题 (每题 25 分, 共 50 分)

题号	1	2
答案		

1. 【问题描述】

按规则计算出 1 到 N 之间的结果:

- (1) 计算形如: $1+2-3+4-5+6-7\dots$
- (2) 程序运行时先输入一个正整数 N;
- (3) 计算时包含输入的 N。如输入 5, 则计算 $1+2-3+4-5$ 的值; 如输入 6, 则计算 $1+2-3+4-5+6$;
- (4) 仅输出计算结果, 不输出其他。

【输入描述】

程序运行时, 输入一个正整数 N。假设输入仅为大于 1 的正整数, 不考虑带小数点的数或负数等不合规情形;

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】

输出形如:

3

输出只能有计算结果, 不得有其他。

【样例输入 1】

2

【样例输出 1】

3

【样例输入 2】

3

【样例输出 2】

0

【参考代码】

#编程题 1, 计算 $1+2-3+4-5+6+\dots$

```
N = int( input() )
SUM = 1
for i in range( 2 , N + 1 ):
    if i % 2 == 0:
        SUM += i
    else:
        SUM -= i
print( SUM )
```

2. 【问题描述】

- (1) 利用 `input()` 语句先后输入两个直角三角形的两个直角边的边长；
- (2) 如果第一个直角三角形的两个直角边比值等于另外一个直角三角形两个直角边的比值，则视为相似；
- (3) 如果相似则输出一个英文半角字母 T，否则输出一个英文半角字母 F；

【输入描述】

- (1) 先后四次输入正整数，前两个为第一个直角三角形的两个直角边，后两个为第二个直角三角形的直角边；
- (2) 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) 如果两个直角三角形相似，则输出一个英文半角字母 T；
- (2) 如果两个直角三角形不相似，则输出一个英文半角字母 F；

【样例输入 1】

1
2

3

6

【样例输出 1】

T

【样例输入 2】

1

2

6

3

【样例输出 2】

T

【参考代码】

```
a=int(input())
```

```
b=int(input())
```

```
c=int(input())
```

```
d=int(input())
```

```
if a * d == c * b or a * c == b * d:
```

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

```
else:
```

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