

## 《Python 程序设计》题库（普及类）

### 基础知识

- 1、Python 标准库 `math` 中用来计算平方根的函数是\_\_\_\_\_。(sqrt)
- 2、Python 程序文件扩展名主要有\_\_和\_\_两种，其中后者常用于 GUI 程序。(py、pyw)
- 3、在 IDLE 交互模式中浏览上一条语句的快捷键是\_\_\_\_\_。(Alt+P)
- 4、在 IDLE 交互模式中执行的快捷键是\_\_\_\_\_。(F5)
- 5、在 Python 中\_\_\_\_\_表示空类型。(None)
- 6、Python 运算符中用来计算整商的是\_\_\_\_\_。(//)
- 7、表达式 `15 // 4` 的值为\_\_\_\_\_。(3)
- 8、Python 运算符中用来余数的是\_\_\_\_\_。(%)
- 9、表达式 `int('123', 16)` 的值为\_\_\_\_\_。(291)
- 10、表达式 `int('123', 8)` 的值为\_\_\_\_\_。(83)
- 11、表达式 `int('123')` 的值为\_\_\_\_\_。(123)
- 12、表达式 `int('101', 2)` 的值为\_\_\_\_\_。(5)
- 13、表达式 `abs(-3)` 的值为\_\_\_\_\_。(3)
- 14、Python 3.x 语句 `print(1, 2, 3, sep=':')` 的输出结果为\_\_\_\_\_。(1:2:3)
- 15、表达式 `int(4**0.5)` 的值为\_\_\_\_\_。(2)
- 16、已知 `x = 3`，那么执行语句 `x += 6` 之后，`x` 的值为\_\_\_\_\_。(9)
- 17、表达式 `3 | 5` 的值为\_\_\_\_\_。(7)
- 18、表达式 `3 & 6` 的值为\_\_\_\_\_。(2)
- 19、表达式 `3 ** 2` 的值为\_\_\_\_\_。(9)
- 20、表达式 `3 * 2` 的值为\_\_\_\_\_。(6)
- 21、表达式 `print(0b10101)` 的值为\_\_\_\_\_。(21)

### Python 序列

- 22、表达式 `[1, 2, 3]*3` 的执行结果为\_\_\_\_\_。( [1, 2, 3, 1, 2, 3, 1, 2, 3] )
- 23、语句 `x = 3==3, 5` 执行结束后，变量 `x` 的值为\_\_\_\_\_。( (True, 5) )
- 24、已知 `x = 3`，那么执行语句 `x *= 6` 之后，`x` 的值为\_\_\_\_\_。(18)
- 25、使用列表推导式生成包含 10 个数字 5 的列表，语句可以写为。( [5 for i in range(10)] )
- 26、Python 语句 `list(range(1,10,3))` 执行结果为\_\_\_\_\_。( [1, 4, 7] )
- 27、表达式 `list(range(5))` 的值为\_\_\_\_\_。( [0, 1, 2, 3, 4] )
- 28、切片操作 `list(range(6))[::2]` 执行结果为\_\_\_\_\_。( [0, 2, 4] )
- 29、已知 `x = list(range(10))`，则表达式 `x[-4:]` 的值为\_\_\_\_\_。( [6, 7, 8, 9] )
- 30、已知 `x=3` 和 `y=5`，执行语句 `x, y = y, x` 后 `x` 的值是\_\_\_\_\_。(5)
- 31、字典中多个元素之间使用\_\_\_\_\_分隔开，每个元素的“键”与“值”之间使用\_\_\_\_\_分隔开。(逗号、冒号)
- 32、已知 `x = {1:2}`，那么执行语句 `x[2] = 3` 之后，`x` 的值为\_\_\_\_\_。( {1:2, 2:3} )
- 33、表达式 `{1, 2, 3, 4} - {3, 4, 5, 6}` 的值为\_\_\_\_\_。( {1, 2} )
- 34、表达式 `set([1, 1, 2, 3])` 的值为\_\_\_\_\_。( {1, 2, 3} )
- 35、使用列表推导式得到 100 以内所有能被 13 整除的数的代码可以写作  
\_\_\_\_\_。( [i for i in range(100) if i%13==0] )
- 36、表达式 `[x for x in [1,2,3,4,5] if x<3]` 的值为\_\_\_\_\_。( [1, 2] )
- 37、已知 `x = [3, 5, 7]`，那么执行语句 `x[1:] = [2]` 之后，`x` 的值为\_\_\_\_\_。( [3, 2] )

38、已知  $x = [3, 5, 7]$ ，那么执行语句  $x[:3] = [2]$  之后， $x$  的值为\_\_\_\_\_。( [ 2] )

39、已知  $x$  为非空列表，那么执行语句  $y = x[:]$  之后， $\text{id}(x[0]) == \text{id}(y[0])$  的值为\_\_\_\_\_。( True )

40、表达式  $[1] * 2$  的值为\_\_\_\_\_。( [1, 1] )

41、表达式  $[1, 2] * 2$  的值为\_\_\_\_\_。( [1, 2, 1, 2] )

42、表达式  $\{1, 2, 3, 4, 5\} \wedge \{4, 5, 6, 7\}$  的值为\_\_\_\_\_。( {1, 2, 3, 6, 7} )

43、表达式  $\{1, 2, 3\} \mid \{3, 4, 5\}$  的值为\_\_\_\_\_。( {1, 2, 3, 4, 5} )

44、表达式  $\{1, 2, 3\} \mid \{2, 3, 4\}$  的值为\_\_\_\_\_。( {1, 2, 3, 4} )

45、表达式  $\{1, 2, 3\} \& \{3, 4, 5\}$  的值为\_\_\_\_\_。( {3} )

46、表达式  $\{1, 2, 3\} \& \{2, 3, 4\}$  的值为\_\_\_\_\_。( {2, 3} )

47、表达式  $\{1, 2, 3\} - \{3, 4, 5\}$  的值为\_\_\_\_\_。( {1, 2} )

48、表达式  $\{1, 2, 3\} < \{3, 4, 5\}$  的值为\_\_\_\_\_。( False )

49、表达式  $\{1, 2, 3\} < \{1, 2, 4\}$  的值为\_\_\_\_\_。( False )

50、表达式  $(1, 2, 3) + (4, 5)$  的值为\_\_\_\_\_。( (1, 2, 3, 4, 5) )

51、表达式  $[i \text{ for } i \text{ in range}(10) \text{ if } i > 8]$  的值为\_\_\_\_\_。( [9] )

52、表达式  $\text{int}('11', 2)$  的值为\_\_\_\_\_。( 3 )

53、表达式  $\text{int}('11', 8)$  的值为\_\_\_\_\_。( 9 )

54、表达式  $\text{int}(\text{bin}(54321), 2)$  的值为\_\_\_\_\_。( 54321 )

55、表达式  $\text{chr}(\text{ord}('A') + 1)$  的值为\_\_\_\_\_。( 'B' )

56、表达式  $\text{int}(\text{str}(34)) == 34$  的值为\_\_\_\_\_。( True )

57、表达式  $\text{list}(\text{str}([3, 4])) == [3, 4]$  的值为\_\_\_\_\_。( False )

58、表达式  $\{1, 2, 3, 4, 5, 6\} \wedge \{5, 6, 7, 8\}$  的值为\_\_\_\_\_。( {1, 2, 3, 4, 7, 8} )

### 选择结构与循环结构

59、表达式  $'ab' \text{ in } 'acbed'$  的值为\_\_\_\_\_。( False )

60、表达式  $'ab' \text{ in } 'abcde'$  的值为\_\_\_\_\_。( True )

61、关键字\_\_\_\_\_用于测试一个对象是否是一个可迭代对象的元素。( in )

62、表达式  $3 < 5 > 2$  的值为\_\_\_\_\_。( True )

63、表达式  $1 < 2 < 3$  的值为\_\_\_\_\_。( True )

64、表达式  $3 \text{ or } 5$  的值为\_\_\_\_\_。( 3 )

65、表达式  $0 \text{ or } 5$  的值为\_\_\_\_\_。( 5 )

66、表达式  $3 \text{ and } 5$  的值为\_\_\_\_\_。( 5 )

67、表达式  $3 \text{ and not } 5$  的值为\_\_\_\_\_。( False )

68、Python 中用于表示逻辑与、逻辑或、逻辑非运算的关键字分别是\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_。( and、or、not )

69、Python 3.x 语句  $\text{for } i \text{ in range}(3): \text{print}(i, \text{end=','})$  的输出结果为\_\_\_\_\_。( 0,1,2, )

70、Python 3.x 语句  $\text{print}(1, 2, 3, \text{sep=','})$  的输出结果为\_\_\_\_\_。( 1,2,3 )

71、对于带有 else 子句的 for 循环和 while 循环，当循环因循环条件不成立而自然结束时\_\_\_\_\_ (会? 不会?) 执行 else 中的代码。( 会 )

72、在循环语句中，\_\_\_\_\_ 语句的作用是提前结束本层循环。( break )

73、在循环语句中，\_\_\_\_\_ 语句的作用是提前进入下一次循环。( continue )

74、表达式  $5 \text{ if } 5 > 6 \text{ else } (6 \text{ if } 3 > 2 \text{ else } 5)$  的值为\_\_\_\_\_。( 6 )

75、Python 关键字 elif 表示\_\_\_\_\_ 和 \_\_\_\_\_ 两个单词的缩写。( else、if )

76、表达式  $3 \text{ in } \{1, 2, 3\}$  的值为\_\_\_\_\_。( True )

77、表达式  $'ac' \text{ in } 'abce'$  的值为\_\_\_\_\_。( False )

78、表达式 `not 3` 的值为\_\_\_\_\_。(False)

### 字符串与正则表达式

79、表达式 `'abc' in ('abcdefg')` 的值为\_\_\_\_\_。(True)

80、表达式 `'abc' in ['abcdefg']` 的值为\_\_\_\_\_。(False)

81、转义字符 `r'\n'` 的含义是\_\_\_\_\_。(回车换行)

82、已知列表对象 `x = ['11', '2', '3']`，则表达式 `max(x)` 的值为\_\_\_\_\_。('3')

83、表达式 `min(['11', '2', '3'])` 的值为\_\_\_\_\_。('11')

84、已知列表对象 `x = ['11', '2', '3']`，则表达式 `max(x, key=len)` 的值为\_\_\_\_\_。('11')

85、表达式 `str([1, 2, 3])` 的值为\_\_\_\_\_。('[1, 2, 3]')

86、表达式 `str((1, 2, 3))` 的值为\_\_\_\_\_。('(1, 2, 3)')

87、表达式 `sum(range(1, 10, 2))` 的值为\_\_\_\_\_。(25)

88、表达式 `sum(range(1, 10))` 的值为\_\_\_\_\_。(45)

89、表达式 `'Hello world'.upper()` 的值为\_\_\_\_\_。('HELLO WORLD')

90、表达式 `'Hello world'.lower()` 的值为\_\_\_\_\_。('hello world')

91、表达式 `'Hello world'.lower().upper()` 的值为\_\_\_\_\_。('HELLO WORLD')

92、表达式 `'a' + 'b'` 的值为\_\_\_\_\_。('ab')

93、已知 `x = '123'` 和 `y = '456'`，那么表达式 `x + y` 的值为\_\_\_\_\_。('123456')

94、表达式 `'Hello world!)[-4]` 的值为\_\_\_\_\_。('r')

95、表达式 `'Hello world!)[-4:]` 的值为\_\_\_\_\_。('rld!')

96、当在字符串前加上小写字母 `r` 或大写字母 `R` 表示原始字符串，不对其中的任何字符进行转义。(r、R)

97、表达式 `'Hello world!'.count('l')` 的值为\_\_\_\_\_。(3)

98、已知 `x = 'abcdefg'`，则表达式 `x[3:] + x[:3]` 的值为\_\_\_\_\_。('defgabc')

### 函数定义与使用

99、Python 中定义函数的关键字是\_\_\_\_\_。(def)

100、在函数内部可以通过关键字\_\_\_\_\_来定义全局变量。(global)

101、如果函数中没有 `return` 语句或者 `return` 语句不带任何返回值，那么该函数的返回值为\_\_\_\_\_。(None)

102、表达式 `sum(range(10))` 的值为\_\_\_\_\_。(45)

103、表达式 `sum(range(1, 10, 2))` 的值为\_\_\_\_\_。(25)

104、表达式 `list(range(50, 60, 3))` 的值为\_\_\_\_\_。([50, 53, 56, 59])

### 判断题

### 基础知识

1、Python 是一种跨平台、开源、免费的高级动态编程语言。(对)

2、Python 3.x 完全兼容 Python 2.x。(错)

3、Python 3.x 和 Python 2.x 唯一的区别就是：`print` 在 Python 2.x 中是输出语句，而在 Python 3.x 中是输入函数。(错)

4、在 Windows 平台上编写的 Python 程序无法在 Unix 平台运行。(错)

5、不可以同一台计算机上安装多个 Python 版本。(错)

6、已知 `x = 3`，那么赋值语句 `x = 'abcdefg'` 是无法正常执行的。(错)

7、Python 变量使用前必须先声明，并且一旦声明就不能再当前作用域内改变其类型。(错)

8、Python 采用的是基于值得自动内存管理方式。(对)

9、Python 不允许使用关键字作为变量名，允许使用内置函数名作为变量名，但这会改变函

数名的含义。(对)

- 10、在 Python 中可以使用 if 作为变量名。(错)
- 11、在 Python 3.x 中可以使用中文作为变量名。(对)
- 12、Python 变量名必须以字母或下划线开头，并且区分字母大小写。(对)
- 13、加法运算符可以用来连接字符串并生成新字符串。(对)
- 14、9999\*\*9999 这样的命令在 Python 中无法运行。(错)
- 15、3+4j 不是合法的 Python 表达式。(错)
- 16、0o12f 是合法的八进制数字。(错)
- 17、不管输入什么，Python 3.x 中 input() 函数的返回值总是字符串。(对)
- 18、pip 命令也支持扩展名为.whl 的文件直接安装 Python 扩展库。(对)
- 19、只有 Python 扩展库才需要导入以后才能使用其中的对象，Python 标准库不需要导入即可使用其中的所有对象和方法。(错)
- 20、在 Python 中 0xad 是合法的十六进制数字表示形式。(对)
- 21、在 Python 中 0oa1 是合法的八进制数字表示形式。(错)
- 22、Python 使用缩进来体现代码之间的逻辑关系。(对)
- 23、Python 代码的注释只有一种方式，那就是使用#符号。(错)
- 24、放在一对三引号之间的任何内容将被认为是注释。(错)
- 25、为了让代码更加紧凑，编写 Python 程序时应尽量避免加入空格和空行。(错)
- 26、在 Python 中可以使用 for 作为变量名。(错)
- 27、Python 关键字不可以作为变量名。(对)
- 28、一个数字 5 也是合法的 Python 表达式。(对)
- 29、不同版本的 Python 不能安装到同一台计算机上。(错)
- 30、Python 变量名区分大小写，所以 student 和 Student 不是同一个变量。(对)
- 31、在 Python 3.x 中，使用内置函数 input() 接收用户输入时，不论用户输入的什么格式，一律按字符串进行返回。(对)

#### 编程题

- 1、编写一个 python 程序，输入两个数，比较它们的大小并输出其中较大者。

```
x=input("输入第一个数：")
y=input("输入第二个数：")
if(x==y):
    print("两数相同")
elif(x>y):
    print(x)
else:
    print(y)
```
- 2、编写程序，在 D 盘根目录下创建一个文本文件 test.txt，并向其中写入字符串 hello world。  
答：

```
fp = open(r'D:\test.txt', 'a+')
print('hello world', file=fp)
fp.close()
```
- 3、编写程序，运行后用户输入 4 位整数作为年份，判断其是否为闰年。如果年份能被 400 整除，则为闰年；如果年份能被 4 整除但不能被 100 整除也为闰年。  
答：

```
x = input('Please input an integer of 4 digits meaning the year:')
```

```
x = eval(x)
if x%400==0 or (x%4==0 and not x%100==0):
    print('Yes')
```

else:

```
    print('No')
```

4、下面程序的执行结果是\_\_\_\_\_。（1275）

```
s = 0
```

```
for i in range(1,101):
```

```
    s += i
```

```
    if i == 50:
```

```
        print(s)
```

```
        break
```

```
else:
```

```
    print(i)
```

6、用 python 定义一个函数，输入一年份，判断该年份是否是闰年并输出结果。

```
year=int(input('请输入一个年份'))
```

```
if year%4==0 and year%100!=0 or year%100==0:
```

```
    print ("是闰年")
```

```
else:
```

```
    print("不是闰年")
```

7、存在字符串“ab2b3n5n2n67mm4n2”，编程统计字符串中字母 n 出现的次数

```
s='ab2b3n5n2n67mm4n2'
```

```
Count=0
```

```
for i in range(0,17):
```

```
    if s[i]=="n":
```

```
        Count += 1
```

```
print (Count)
```