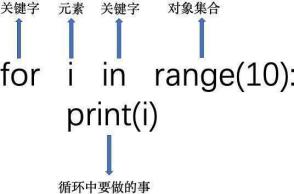
一、单项选择题（共 20 题，每题 2.5 分，共 50 分）

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 题号 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 答案 | A | D | A | D | B | B | D | C | B | C |
| 题号 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| 答案 | D | A | A | B | A | D | A | D | B | C |

1. 解析：首先给变量 b 赋值 2，变量 a 赋值 3，

计算 b+=a，也就是变量 b = b+a=2+3, 此时变量 b 等于 5； 计算 b\*=a, 也就是变量 b = b\*a=5\*3，此时变量 b 等于 15； 计算 b/=a, 也就是变量 b = b/a=15/3，此时变量 b 等于 5.0； 在 Python 运算中，需要注意的是除法运算的结果是浮点型。

1. 解析：Python for 循环可以遍历任何序列的项目，如一个列表或者一个字符串。比如循环 10 次，就是 for i in range(10)。



1. 解析：选项 B，if 条件后面缺少一个冒号；

选项 C，大于等于符号正确的书写形式为'>=' ；

选项 D，冒号的位置是在 if 条件后面，不是 if 后面。

1. 解析：选项 A、B、C 依次是加法赋值运算符、减法赋值运算符、乘法赋值运算符。选项 D 是无效语法， 在 Python 中没有这样的编程语法。
2. 解析：这是一道编程题。input() 函数接受一个标准输入数据，返回为 string 类型，int()函数把字符串型数字转换成整型数字储存在变量 a 中。接着是 if 条件判断语句，通过 if 语句中的 print 语句，我们可以知道，如果 if 条件成立，就会打印的内容是变量 a 能同时被 3、5、7 整除。要让条件同时成立，我们需要用到逻辑运算符 and。
3. 解析：在 Python 中，'\*' 表示两个数相乘或是返回一个被重复若干次的字符串；‘\*\*’表示返回 x 的 y 次幂， 例如：2\*\*3 就是计算 2 的 3 次方，也就是数字 8。
4. 解析：在 Python 中，‘//’表示取整除，即向下取接近商的整数。
5. 解析：在 Python 中，'\*' 表示两个数相乘或是返回一个被重复若干次的字符串；‘\*\*’表示返回 x 的 y 次幂。在 Python 中运算符是符合四则运算计算规则的，\*\* 冥运算的优先级是最高的，所以运算式 3\*1\*\*3 = 3\*(1\*\*3) = 3。
6. 解析：while 循环语句，while 循环通过一个能够产生或转换出 bool 值的表达式来控制循环。

选项 A、选项 C 缺少冒号，选项 D 中的 loop 是无效语法。

1. 解析：Python 中的变量不需要声明。每个变量在使用前都必须赋值，变量赋值以后该变量才会被创建。等号（=）运算符左边是一个变量名,等号（=）运算符右边是存储在变量中的值。
2. 解析：Python 中单行注释以 # 开头。选项 A 是乘法运算符，选项 C 是取整除运算符。
3. 解析：题目要求完成数值交换，假设 x 存储的输入值为 5，y 存储的输入值为 10，那么要求的输出结果就是 x 为 10，y 为 5.

选项 B，la = x，x = y，y = la，在 Python 中变量名不能用数字开头，程序会报错； 选项 C，x = y，y = n，n = x，运行到 y=n 程序会报错，错误提示为变量 n 未赋值； 选项 D，x=c，c=y，y=x，运行到 x = c 程序会报错，错误提示为变量 c 未赋值。

1. 解析：在 Python 中，创建一个列表，只要把逗号分隔的不同的数据项使用方括号括起来即可。创建一个字典，字典的每个键值对用冒号分割，每个对之间用逗号分割，整个字典包括在花括号中。创建一个元 祖，只需要在括号中添加元素，并使用逗号隔开即可。

选项 D 不是 Python 的数据类型

1. 解析：在 Python 中，‘%’表示取模运算，返回除法的余数。22 对 3 取模的商是 7，余数是 1。
2. 解析：在 Python 中多行注释可以用多个 # 号，还有 ''' 和 """。
3. 解析：Python3 中有六个标准的数据类型：Number（数字）、String（字符串）、List（列表）、Tuple

（元组）、Set（集合）、Dictionary（字典）。类不是数据类型。

1. 解析：在 Python 中，% 被称为格式化操作符，专门用于处理字符串中的格式，包含%的字符串，被称为格式化字符串。 %f 默认小数点后六位，%.2f 表示小数点后只显示两位。
2. 解析：在 Python 中，创建一个列表，只要把逗号分隔的不同的数据项使用方括号括起来即可。
3. 解析：int() 函数是 Python 的内置函数，用于将一个字符串或数字转换为整型。
4. 解析：if 语句添加其他条件语句即为多分支结构，分支结构为 if-elif-else 条件判断语句。

二、多项选择题（共 5 题，每题 2 分，共 10 分）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 题号 | 21 | 22 | 23 | 24 | 25 |
| 答案 | ABCD | ABC | AC | BCD | ABD |

1. 解析：算术运算符，例如：‘+’加法运算符；关系运算符，例如：‘==’等于，‘>’大于；逻辑运算符，例如：‘and' 布尔"与"；赋值运算符，例如：‘+='加法赋值运算符。
2. 解析：Python3 中有六个标准的数据类型：Number（数字）、String（字符串）、List（列表）、Tuple

（元组）、Set（集合）、Dictionary（字典）。Number（数字）中包含整数型（int)和浮点型（float)等。选项 D，Tuple 单词多写了一个字母 s。

1. 解析：选项 A，str1.index(str2, beg=0, end=len(str1))，可用于检测字符是否包含在字符串中，如果检测

的字符不在字符串中，程序会报错；

选项 B，str.count(sub, start= 0,end=len(string))，返回 str 在 string 里面出现的次数，如果 beg 或者 end 指定则返回指定范围内 str 出现的次数；

选项 C，str1.find(str2, beg=0, end=len(string))，检测 str 2 是否包含在字符串中，如果指定范围 beg 和 end ， 则检查是否包含在指定范围内，如果包含返回开始的索引值，否则返回-1

选项 D，replace(str1, str2)方法至少需要填入 2 个参数，可以将字符串中的字符 str1 替换成 str2.

1. 解析：关系运算符也称为比较运算符，选项 A 是简单的赋值运算符。
2. 解析：Python 逻辑运算符只有 3 种：或（or）、与（and）、非（not)。

三、判断题（共 10 题，每题 1 分，共 10 分）

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 题号 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 |
| 答案 | T | F | T | F | T | T | T | T | F | F |

1. 解析：相对于编译型语言存在的，源代码不是直接翻译成机器语言，而是先翻译成中间代码，再由解 释器对中间代码进行解释运行。比如 Python/JavaScript / Perl /Shell 等都是解释型语言。
2. 解析：多项分支的判断顺序是自上而下逐个分支进行判断。需要注意的是 elif 可以有很多个，分支中

else 可有可无，多分支只会选一个执行。

29. 解析：布尔类型（bool 类型）是计算机中最基本的类型，它是计算机二进制世界的体现，一切都是 0

和 1。Python 中的布尔类型只有两种值：True 和 False。

1. 解析：Python 中 if 条件循环可以有 else 语句，异常处理 try/except 也可以有 else 语句。
2. 解析：在字符串中，加号用于拼接字符串。a\*2 输出结果为:HelloHello。

四、编程题（共 2 题，每题 15 分，共 30 分）

1. 参考代码：

mount\_hight = 8848000

count = 0

hight = 0.1

while hight < mount\_hight: hight \*= 2

count += 1 print(count)

输入打印的结果为：27

1. 参考代码

hight1 = 100

hight2 = 50

for i in range(1,5):` hight1 += 2\*hight2 hight2 /= 2

print(hight1, hight2)

打印的结果为：287.5 3.125