错了的题号我都用红字黄底标出来啦，你再看看。

一、单选题(共25题，每题2分，共50分)

|  |  |
| --- | --- |
| 1. | 下面程序执行完毕后，最终的结果是？（ ）  a=[34,17,7,48,10,5] b=[] c=[] while len(a)>0:     s=a.pop()     if(s%2==0):         b.append(s)     else:         c.append(s) print(b) print(c) |
|  | |  |  | | --- | --- | | A. | [34, 48,10] [17, 7, 5] |  |  |  | | --- | --- | | B. | [10, 48, 34] [5, 7, 17] |  |  |  | | --- | --- | | C. | [10, 48, 34] [17, 7, 5] |  |  |  | | --- | --- | | D. | [34, 48, 10] [5, 7, 17] | | B | 考点：%取余数，可变序列的通用操作之列表的 .pop(),.append()方法  .pop()、.append()方法默认都是对最后一个位置进行执行操作  区分方法和函数，.pop()、.append()都是方法，len()是函数。  技巧：判断一个5一个10就出来答案了，不放心的话可以继续往后看 | |

|  |  |
| --- | --- |
| 2. | 以下程序的运行结果是？（ ）  l =["兰溪","金华","武义","永康","磐安","东阳","义乌","浦江"] for s in l:     if"义"in s:       print(s) |
|  | |  |  | | --- | --- | | A. | 兰溪 金华 武义 |  |  |  | | --- | --- | | B. | 武义 义乌 |  |  |  | | --- | --- | | C. | 武义 |  |  |  | | --- | --- | | D. | 义乌 | | B | 考点： 序列的通用操作之判断元素是否在序列内  粗心错的 | |

|  |  |
| --- | --- |
| 3. | 以下程序的输出结果是？（ ）  ls = [1,2,3] lt = [4,5,6] print(ls+lt) |
|  | |  |  | | --- | --- | | A. | [1,2,3,4,5,6] |  |  |  | | --- | --- | | B. | [1,2,3,[4,5,6]] |  |  |  | | --- | --- | | C. | [4,5,6] |  |  |  | | --- | --- | | D. | [5,7,9] | | A | 考点：序列的通用操作之拼接 | |

|  |  |
| --- | --- |
| 4. | 列表listV = list(range(10)),以下能够输出列表listV中最小元素的是？（ ） |
|  | |  |  | | --- | --- | | A. | print(min(listV)) |  |  |  | | --- | --- | | B. | print(listV.max( )) |  |  |  | | --- | --- | | C. | print(min(listV())) |  |  |  | | --- | --- | | D. | print(listV.revrese(i)[0]) | | A | 考点：range,类型转化，序列的通用函数  Range(10)生成了一个range类型的对象，list()转化成list类型，min()函数，  找最小值，Min()属于序列的通用函数。  B：listV.max()这是方法的用法， min max 是函数  D：.reverse()方法是反向列表中的元素，没有传参， 可变序列的通用操作。 | |

|  |  |
| --- | --- |
| 5. | 以下程序的输出结果是（ ）。  a = tuple('abcdefg')  print(a) |
|  | |  |  | | --- | --- | | A. | ('a', 'b', 'c', 'd', 'e', 'f', 'g') |  |  |  | | --- | --- | | B. | ['a', 'b', 'c', 'd', 'e', 'f', 'g'] |  |  |  | | --- | --- | | C. | ['abcdefg'] |  |  |  | | --- | --- | | D. | 'abcdefg' | | A | 考点：类型转化  此乃将字符串转化为元组，字符串中每一个字符都将是元组中的一个元素。  Tuple就是元组的意思 | |

|  |  |
| --- | --- |
| 6. | 运行如下程序，结果是？（ ）  l=[1,"laowang",3.14,"laoli"] l[0]=2 del l[1] print(l) |
|  | |  |  | | --- | --- | | A. | [1, 3.14, 'laoli'] |  |  |  | | --- | --- | | B. | [2, 3.14, 'laoli'] |  |  |  | | --- | --- | | C. | ["laowang",3.14, 'laoli'] |  |  |  | | --- | --- | | D. | [2,"laowang",3.14,] | | B | 考点： del 可变序列的通用操作 | |

|  |  |
| --- | --- |
| 7. | 关于列表s的相关操作，描述不正确的是？（ ） |
|  | |  |  | | --- | --- | | A. | s.append():在列表末尾添加新的对象 |  |  |  | | --- | --- | | B. | s.reverse():反转列表中的元素 |  |  |  | | --- | --- | | C. | s.count():统计某个元素在列表中出现的次数 |  |  |  | | --- | --- | | D. | s.clear()：删除列表s的最后一个元素 | | D |  | |

|  |  |
| --- | --- |
| 8. | 关于以下代码，描述正确的是？（ ） a = 'False' if a:     print('True') |
|  | |  |  | | --- | --- | | A. | 上述代码的输出结果为True。 |  |  |  | | --- | --- | | B. | 上述代码的输出结果为False。 |  |  |  | | --- | --- | | C. | 上述代码存在语法错误。 |  |  |  | | --- | --- | | D. | 上述代码没有语法错误，但没有任何输出。 | | A | 首先a是一个字符串，且是一个非空字符串，所以他为True  其次， 字符串的内容是False，这完全是障眼法，不用管内容是什么。  If 是判断 T OR F 的， 只要字符串不空，都是True。 | |

|  |  |
| --- | --- |
| 9. | 下列代码的输出结果是？（ ）  ls = [[0,1],[5,6],[7,8]]  lis = []  for i in range(len(ls)):       lis.append(ls[i][1])  print(lis) |
|  | |  |  | | --- | --- | | A. | [1,6,8] |  |  |  | | --- | --- | | B. | [0,5,7] |  |  |  | | --- | --- | | C. | [0,6,8] |  |  |  | | --- | --- | | D. | [0,1] | | A | 考点：range,序列的通用操作len，获得序列的长度，可变序列的通用操作  Append方法。  Len(ls)结果是3，i 取值 0，1，2， append的值分别是ls[0][1], 1 1, 2 1 | |

|  |  |
| --- | --- |
| 10. | 已知列表lis=['1','2',3]，则执行print(2 in lis)语句输出的结果是？（ ） |
|  | |  |  | | --- | --- | | A. | True |  |  |  | | --- | --- | | B. | true |  |  |  | | --- | --- | | C. | False |  |  |  | | --- | --- | | D. | false | | C | 考点：序列的通用操作之 in， not in  2 是 int 型， ‘2’ 是str型，不是一回事。 | |

|  |  |
| --- | --- |
| 11. | 现在有s="abcdefghi"，请问s[4]的值是？（ ） |
|  | |  |  | | --- | --- | | A. | d |  |  |  | | --- | --- | | B. | e |  |  |  | | --- | --- | | C. | abcd |  |  |  | | --- | --- | | D. | 0 | | B | 考点：序列的通用操作之字符串通过下标访问元素。 | |

|  |  |
| --- | --- |
| 12. | 下面代码的输出结果是？（ ） a={'sx':90,'yuwen':93,'yingyu':88,'kexue':98} print(a['sx']) |
|  | |  |  | | --- | --- | | A. | 93 |  |  |  | | --- | --- | | B. | 90 |  |  |  | | --- | --- | | C. | 88 |  |  |  | | --- | --- | | D. | 98 | | B | 字典的常用操作之访问 | |

|  |  |
| --- | --- |
| 13. | 下面代码的输出结果是？（ ）  a=[1,3,5,7,9] for i in a:     print(i) |
|  | |  |  | | --- | --- | | A. | 1,3,5,7,9 |  |  |  | | --- | --- | | B. | [1,3,5,7,9] |  |  |  | | --- | --- | | C. | 1 3 5 7 9 |  |  |  | | --- | --- | | D. | 9 7 5 3 1 | | C | 考点：序列的通过用操作之 in not in  注意：元素之间没有逗号（其实是每个数字一行，在这里不能太较真） | |

|  |  |
| --- | --- |
| 14. | 以下用于Python循环结构的关键字是（ ） |
|  | |  |  | | --- | --- | | A. | while |  |  |  | | --- | --- | | B. | loop |  |  |  | | --- | --- | | C. | if |  |  |  | | --- | --- | | D. | do…for | | A |  | |

|  |  |
| --- | --- |
| 15. | 以下代码绘制的图形是？（  ） import turtle for i in range(1,7):     turtle.fd(50)     turtle.left(60) |
|  | |  |  | | --- | --- | | A. | 正方形 |  |  |  | | --- | --- | | B. | 六边形 |  |  |  | | --- | --- | | C. | 三角形 |  |  |  | | --- | --- | | D. | 五角星 | | B | 考点：range 1到6 | |

|  |  |
| --- | --- |
| 16. | 已知列表a=[1,2,3]，b=['4']，执行语句print(a+b)后，输出的结果是？（ ） |
|  | |  |  | | --- | --- | | A. | [1,2,3,4] |  |  |  | | --- | --- | | B. | [1,2,3,'4'] |  |  |  | | --- | --- | | C. | ['1','2','3','4'] |  |  |  | | --- | --- | | D. | 10 | | B | 考点：序列的通用操作之连接序列。  不改变原序列中元素类型。 | |

|  |  |
| --- | --- |
| 17. | 已知列表a=[1,2,3,4,5]，下列语句输出结果为False的是？（ ） |
|  | |  |  | | --- | --- | | A. | print(a[3]==a[-2]) |  |  |  | | --- | --- | | B. | print(a[:3]==a[:-2]) |  |  |  | | --- | --- | | C. | print(a[:1]==a[0]) |  |  |  | | --- | --- | | D. | print(a[2]==a[-3]) | | C | 考点：  访问指定索引范围（切片访问）的结果是一个序列（本题是列表）  访问返回的结果是序列中的元素。  切片访问默认值 | |

|  |  |
| --- | --- |
| 18. | 在python中，表示跳出当前循环的语句是？（ ） |
|  | |  |  | | --- | --- | | A. | break |  |  |  | | --- | --- | | B. | pass |  |  |  | | --- | --- | | C. | exit |  |  |  | | --- | --- | | D. | Esc | | A |  | |

|  |  |
| --- | --- |
| 19. | 已知字典score={"语文":95,"数学":93,"英语":97},  则执行print(score["语文"]+score["数学"]//2)，输出的结果为？（ ） |
|  | |  |  | | --- | --- | | A. | 141 |  |  |  | | --- | --- | | B. | 141.5 |  |  |  | | --- | --- | | C. | 94 |  |  |  | | --- | --- | | D. | 94.0 | | A | 考点：  字典的常用操作之访问，//是地板除（向下取整）  之前不熟悉这个知识点 | |

|  |  |
| --- | --- |
| 20. | 以下程序的运行结果是？（ ）  a={"name":"jt","age":29,"class":5} a["age"]=15 a["school"]="派森社" print("age:",a["age"]) print("school:",a["school"]) |
|  | |  |  | | --- | --- | | A. | age: 29 school: 派森社 |  |  |  | | --- | --- | | B. | age: 15 |  |  |  | | --- | --- | | C. | age: 15 school: 派森社 |  |  |  | | --- | --- | | D. | school: 派森社 | | C | 考点：  第二句，字典的修改；第三句，字典的增加；第四五句是字典的访问。 | |

|  |  |
| --- | --- |
| 21. | 下列程序的执行结果是？（ ）  s=(1,2,3,4,5,6,7,8)  print(len(s),max(s),min(s)) |
|  | |  |  | | --- | --- | | A. | 7 8 1 |  |  |  | | --- | --- | | B. | 8 8 1 |  |  |  | | --- | --- | | C. | 8 1 8 |  |  |  | | --- | --- | | D. | 7 1 8 | | B | 考点：  序列的通用操作 | |

|  |  |
| --- | --- |
| 22. | 下列不属于Python中处理字典的方法是？（ ） |
|  | |  |  | | --- | --- | | A. | pop() |  |  |  | | --- | --- | | B. | replace() |  |  |  | | --- | --- | | C. | get() |  |  |  | | --- | --- | | D. | popitem() | | B | dict.get(key, default=None)  返回指定键的值，如果值不在字典中返回default值  ####################################  popitem()  返回并删除字典中的最后一对键和值。  ####################################  replace() 方法把字符串中的 old（旧字符串） 替换成 new(新字符串)，如果指定第三个参数max，则替换不超过 max 次。 | |

|  |  |
| --- | --- |
| 23. | 下列语句，不能创建元组的是？（ ） |
|  | |  |  | | --- | --- | | A. | tup=() |  |  |  | | --- | --- | | B. | tup=(1) |  |  |  | | --- | --- | | C. | tup=1,2 |  |  |  | | --- | --- | | D. | tup=(1,2) | | B | tup=(1,)这样才对。 | |

|  |  |
| --- | --- |
| 24. | s="abc123",采用字符串操作函数将其中的字符c替换为字符C，以下哪个操作正确？（ ） |
|  | |  |  | | --- | --- | | A. | s.replace('c', 'C') |  |  |  | | --- | --- | | B. | replace(c,C) |  |  |  | | --- | --- | | C. | s.replace(c,C) |  |  |  | | --- | --- | | D. | replace('abc123','abC123') | | A |  | |

|  |  |
| --- | --- |
| 25. | 下面程序的执行结果为？（ ） s = '{0}+{1}={2}'.format(2, 3, 5) print(s) |
|  | |  |  | | --- | --- | | A. | 0+1=2 |  |  |  | | --- | --- | | B. | {0}+{1}={2} |  |  |  | | --- | --- | | C. | 2+3=5 |  |  |  | | --- | --- | | D. | {2}+{3}={5} | | C | 考点：  Format | |

二、判断题(共10题，每题2分，共20分)

|  |  |
| --- | --- |
| 26. | 已知字符串str='www.baidu.com'，  那么print(str.split('.',1))的结果是：['www', 'baidu','com']。 |
|  | |  |  |  | | --- | --- | --- | | 正确 | 错误 |  | | 错 |  | str.split(str="", num=string.count(str)).  str -- 分隔符，默认为所有的空字符，包括空格、换行(\n)、  制表符(\t)等。返回分割后的字符串列表。  num -- 分割次数。默认为 -1, 即分隔所有。  Python split() 通过指定分隔符对字符串进行切片，如果参数 num 有指定值，则分隔 num+1 个子字符串 | |

|  |  |
| --- | --- |
| 27. | 运行如下程序 a1='hello' a2='everybody' print(a1\*2+a2) 运行结果为hellohelloeverybody |
|  | |  |  |  | | --- | --- | --- | | 正确 | 错误 | 正确 考点：字符串的重复与拼接 | |

|  |  |
| --- | --- |
| 28. | 元组中不可以通过下标索引获取元素。 |
|  | |  |  |  | | --- | --- | --- | | 正确 | 错误 | 错误 | |

|  |  |
| --- | --- |
| 29. | 在使用get（）语句返回字典中指定键的值的时候，如果该键的值在字典中不存在，  则返回系统默认值“unKnown”。 |
|  | |  |  |  | | --- | --- | --- | | 正确 | 错误 | 错误  dict.get(key, default=None)  返回指定键的值，如果值不在字典中返回default值 | |

|  |  |
| --- | --- |
| 30. | 元组是一种可变的序列，创建后可以修改。 |
|  | |  |  |  | | --- | --- | --- | | 正确 | 错误 | 错误 | |

|  |  |
| --- | --- |
| 31. | 条件语句中，if...语句和if...else语句没有区别。 |
|  | |  |  |  | | --- | --- | --- | | 正确 | 错误 | 错误 | |

|  |  |
| --- | --- |
| 32. | 列表是一种序列，列表的元素可以追加、替换、插入和删除。 |
|  | |  |  |  | | --- | --- | --- | | 正确 | 错误 | 正确 | |

|  |  |
| --- | --- |
| 33. | 在循环语句中， break 语句的作用是提前进入下一次循环。 |
|  | |  |  |  | | --- | --- | --- | | 正确 | 错误 | 错误 | |

|  |  |
| --- | --- |
| 34. | 执行以下程序，运行结果是：14  a=1  while a<100:      if a%2==0 and a%7==0:          print(a)          break      a=a+1 |
|  | |  |  |  | | --- | --- | --- | | 正确 | 错误 | 正确 | |

|  |  |
| --- | --- |
| 35. | 语句  for i in range(3) :      print(i, end=',')  的输出结果为0,1,2, |
|  | |  |  |  | | --- | --- | --- | | 正确 | 错误 | 正确 | |

三、编程题(共3题，共30分)

|  |  |
| --- | --- |
| 36. | 数字转汉字  用户输入一个1~9（包含1和9）之间的任一数字，程序输出对应的汉字。  如输入2，程序输出“二”。可重复查询。 |
|  | 试题类型：编程题 |
|  | ****评分标准：****  （1) 数字能够准确地转为汉字;（5分）  （2) 1~9（包含1和9）都能转换;（3分）  （3)  可重复查询.（2分） |
|  | ****参考程序一：****  dd=['一','二','三','四','五','六','七','八','九']  while 1:      a=int(input('输入数字：'))      print(dd[a-1])  ****参考程序二：****  dd=('一','二','三','四','五','六','七','八','九')  while True:      a=int(input('输入数字：'))      print(dd[a-1])  ****参考程序三：****  dd={'1':'一', '2':"二",'3':"三",'4':'四','5':"五",'6':"六",'7':'七','8':"八",'9':"九"} while True:      a=int(input('输入数字：'))      print(dd[str(a)]) |

|  |  |
| --- | --- |
| 37. | 假设10位评委的打分是99,80,86,89,94,92,75,87,86,95，  现需要运用Python语言进行编程实现：  去掉一个最高分，去掉一个最低分，计算平均分，并打印出来。打印格式为：  去掉一个最高分：XX分，去掉一个最低分：XX分，最后得分为：XX分 |
|  | 试题类型：编程题 |
|  | ****评分标准：****  （1) 列表的使用：元素的访问，元素的删除;（3分）  （2) 分值大小的比较;（1分）  （3) 列表元素的遍历;（2分）  （4) 计算总和和平均值;（2分）  （5) 输出正确的结果;（2分） |
|  | b = [99, 80, 86, 89, 94, 92, 75, 87, 86, 95] # 录入10位评委的打分  maxNum = max(b) # 将最高分保存在maxNum变量中  b.remove(maxNum) # 使用remove()函数将最高分去掉  minNum = min(b) # 将最低分保存在minNUm变量中  b.remove(minNum) # 使用remove()函数将最低分去掉  sumNum = sum(b) # 将剩余打分的总和存入sumNum中  avg = sumNum/8 # 求出平均分保存在avg变量中  print('去掉一个最高分：{}分，去掉一个最低分：{}分，最后得分为：{}分'.format(maxNum, minNu |