**2020~2021学年第二学期《Python语言程序设计》期末试卷（A）**

**学院 专业班级 学号 姓名 成绩**

1. **选择题 (用铅笔把选中的方格涂黑 40×1分)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | **A** | **B** | **C** | **D** | **2** | **A** | **B** | **C** | **D** | **3** | **A** | **B** | **C** | **D** | **4** | **A** | **B** | **C** | **D** |
| **5** | **A** | **B** | **C** | **D** | **6** | **A** | **B** | **C** | **D** | **7** | **A** | **B** | **C** | **D** | **8** | **A** | **B** | **C** | **D** |
| **9** | **A** | **B** | **C** | **D** | **10** | **A** | **B** | **C** | **D** | **11** | **A** | **B** | **C** | **D** | **12** | **A** | **B** | **C** | **D** |
| **13** | **A** | **B** | **C** | **D** | **14** | **A** | **B** | **C** | **D** | **15** | **A** | **B** | **C** | **D** | **16** | **A** | **B** | **C** | **D** |
| **17** | **A** | **B** | **C** | **D** | **18** | **A** | **B** | **C** | **D** | **19** | **A** | **B** | **C** | **D** | **20** | **A** | **B** | **C** | **D** |
| **21** | **A** | **B** | **C** | **D** | **22** | **A** | **B** | **C** | **D** | **23** | **A** | **B** | **C** | **D** | **24** | **A** | **B** | **C** | **D** |
| **25** | **A** | **B** | **C** | **D** | **26** | **A** | **B** | **C** | **D** | **27** | **A** | **B** | **C** | **D** | **28** | **A** | **B** | **C** | **D** |
| **29** | **A** | **B** | **C** | **D** | **30** | **A** | **B** | **C** | **D** | **31** | **A** | **B** | **C** | **D** | **32** | **A** | **B** | **C** | **D** |
| **33** | **A** | **B** | **C** | **D** | **34** | **A** | **B** | **C** | **D** | **35** | **A** | **B** | **C** | **D** | **36** | **A** | **B** | **C** | **D** |
| **37** | **A** | **B** | **C** | **D** | **38** | **A** | **B** | **C** | **D** | **39** | **A** | **B** | **C** | **D** | **40** | **A** | **B** | **C** | **D** |

**二、填空题（按空填写，不要按题号填写, 20×2分）**

**1. 2.**

**3. 4.**

**5. 6.**

**7. 8.**

**9. 10.**

**11. 12.**

**13. 14.**

**15. 16.**

**17. 18.**

**19. 20.**

**三、编程题(20分)**

1. **编写程序，输出1000~2000年之间所有的闰年，要求每行输出10个年份。**

**要求：(1).编写函数判断某一年是否为闰年，是闰年返回True,不是闰年返回False；**

**(2).主程序调用函数并输出1000~2000年之间所有的闰年，且每行输出10个年份。**

**（注：闰年是能被4整除但不能被100整除，或能被400整除的年份。）**

**2.编程实现统计一段英文文本每个单词的出现频率，并返回出现频率最高的前5个单词及其出现次数。**

**要求（1）编写函数1实现对文本进行全小写及去标点的预处理；**

1. **编写函数2实现对预处理后的文本统计单词出现频率；**
2. **主程序调用编写的两个函数，实现统计英文文本每个单词的出现频率，并输出出 现频率最高的前5个单词及其出现次数，同时要求单词的输出格式为宽度10，左对齐，空格填充，出现次数的输出格式为右对齐。**

**（注：设待处理的英文以字符串形式赋给变量text='''Hohai University is a research university in Nanjing,China under the direct jurisdiction of the Ministry of Education of China. Hohai University provides internet service all day. Students**

**can find classrooms, dining halls,shopping centers, entertainment center, gym, and swimming pool on campus.'''，在编程中可以直接用text表示）**

**一、选择题 (用铅笔把选中的字母涂黑，共40题，每题1分，共40分)**

1. **下列选项中，不属于Python特点的是（ ）**

**A. 编译性语言 B. 开源理念**

**C. 可移植性 D. 跨平台**

1. **下面哪个不是Python合法的标识符（ ）**

**A. int32 B. 40XL C.self D. \_name\_**

1. **下列哪个语句在Python中是非法的？（ ）**

**A. x=y==1 B.x,y=1 C. x,y=y,x D. x//=y**

1. **以下程序的输出结果是()  
   t="the World is so big,I want to see."**

**s=t[20:21]+’love’+t[:9]  
print(s)**

**A. I love the B. I love World**

**C. I love the World D. I love the Worl**

1. **关于random库，以下选项中描述错误的是（ ）**

**A.生成随机数之前必须要指定随机数种子**

**B.设定相同种子，每次调用随机函数生成的随机数相同**

**C.通过from random import \*可以引入random随机库**

**D.通过import random可以引入random随机库**

1. **下面不是用“+”的用法的是（ ）。**

**A. 字符串连接 B. 算术加法**

**C. 逻辑与 D. 单目运算(正号)**

1. **以下说法中正确的是（ ）。**

**A. break 能结束循环，而continue只能结束本次循环**

**B. break用在for语句中，而continue用在while语句中**

**C. continue用在for语句中，而break用在while语句中**

**D. continue能结束循环，而break只能结束本次循环**

1. **表达式eval("2+4/5" )的执行结果是（ ）。**

**A. 2.8 B. 2 C. 2+4/5 D. "2+4/5"**

1. **以下关于Python语言的条件说法正确的是（ ）。**

**A. 条件12<=5<=8是合法的，输出True**

**B. 条件10<=5<=1是不合法的,抛出异常**

**C. 条件1<=5<=10是合法的，输出False**

**D. 条件3<=8<=10是合法的，等价于3<=8 and 8<=10**

1. **下面代码的输出结果是（ ）。**

**for i in sorted(range(5,11,2),reverse=True):**

**print(i,end=",")**

**A. 5,7,9,11 B. 11,9,7,5**

**C. 5,7,9 D. 9,7,5**

1. **在下列表达式中,（）不是合法的元组。**

**A.(20,) B.('runoob')**

**C.() D.(123,'runoob')**

1. **以下程序的输出结果是（ ）。**

**s='LOVES'  
print('{:\*^13}'.format(s))**

**A. \*\*\*\*\*\*\*\*LOVES B. LOVES\*\*\*\*\*\*\*\***

**C. \*\*\*\*LOVES\*\*\*\* D. LOVES**

1. **以下代码绘制的图形是（ ）。**

**import turtle as t**

**t.left(90)**

**t.forward(200)**

**t.left(90)**

**t.forward(200)**

**t.left(90)**

**t.forward(200)**

**t.left(90)**

**t.forward(200)**

**A. 等边三角形 B. 正方形 C. 正五边形 D. 正六边形**

1. **下面不属于Python标准库的是（ ）。**

**A. wordcloud B. turtle C. random D. math**

1. **执行下面的代码，输出结果是（ ）。**

**ls=[[1,2,3],'python'，[[14,5,'ABC'],6],[7,8]]  
print(ls[0][1])**

**A. 1 B. 14 C. 2 D. 7**

1. **下面不能正确描述[0,100]内的所有能被3整除的数的列表生成式的是（ ）**

**A. [i for i in range(0,100,3)]**

**B. [i for i in range(0,100) if i%3==0]**

**C. [i for i in range(0,100) if not(i%3)]**

**D. [i for i in range(100,0,-3)]**

1. **以下的程序的输出结果是（ ）。**

**def test( b = 2, a = 4):  
      global z  
      z += a \* b  
      return z  
z = 10  
print(z, test())**

**A. 18 None B. 10 18 C.UnboundLocalError D. 18 18**

1. **关于Python的列表,描述错误的选项是（ ）。**

**A. Python列表是包含0个或者多个对象引用的有序序列**

**B. Python列表用中括号[ ]表示**

**C. Python列表是一个可以修改数据项的序列类型**

**D. Python列表的长度是不可变的**

1. **下面代码的输出结果是（ ）。**

**ls =tuple(range(1,4))**

**print(ls)**

**A. [1,2,3,4] B. (1,2,3,4)**

**C. [1,2,3] D. (1,2,3)**

1. **给出下面代码，以下选项中能输出“海贝色”的是（ ）。**

**DictColor = {"seashell":"海贝色","gold":"金色","pink":"粉红色","brown":"棕色", "purple":"紫色","tomato":"西红柿色"}**

**A. print(DictColor.keys())**

**B. print(DictColor["海贝色"])**

**C. print(DictColor.values())**

**D. print(DictColor["seashell"])**

1. **元组变量t=("cat", "dog", "tiger", "human"), t[::-1]的结果是（ ）。**

**A. {'human', 'tiger', 'dog', 'cat'}**

**B. ['human', 'tiger', 'dog', 'cat']**

**C. 运行出错**

**D. ('human', 'tiger', 'dog', 'cat')**

1. **给定字典d,以下选项中对d.items()的描述正确的是（ ）。**

**A.返回一个集合类型,每个元素是一个二元元组,包括字典d中所有键值对**

**B.返回一个列表类型,每个元素是一个二元元组,包括字典d中所有键值对**

**C.返回一个元组类型,每个元素是一个二元元组,包括字典中所有键值对**

**D.返回一种dict\_items类型,包括字典d中所有键值对**

1. **以下不能创建一个集合的语句是（ ）。**

**A. set\_1=set("use") B. set\_1={1,2,3}**

**C. set\_1={} D. set\_1=set([1,2,3])**

1. **以下选项中,不是建立字典的方式是（ ）。**
2. **d = {[1,2]:1, [3,4]:3} B. d = {(1,2):1, (3,4):3}**

**C. d = {'张三':1, '李四':2} D. d = {1:[1,2], 3:[3,4]}**

1. **若字典d= {'Name': 'Kate', 'No': '1001', 'Age': '20', 'Score': '92'}，则len(d)的结果为（ ）。**

**A. 8 B. 9 C. 4 D. 12**

1. **S和T是两个集合,对S^T的描述正确的是（ ）。**

**A.S和T的差运算,包括在集合S但不在T中的元素**

**B.S和T的并运算,包括在集合S和T中的所有元素**

**C.S和T的交运算,包括同时在集合S和T中的元素**

**D.S和T的对称差运算,包括集合S和T中的非相同元素**

1. **以下Python语句的输出结果是（ ）。**

**nums=set([1,2,2,3,3,3,4,4,5])**

**print(len(nums))**

**A. 9 B. 5 C. 4 D. 7**

1. **下列序列属于无序序列的是（ ）。**

**A. 列表 B. 集合**

**C. 元组 D. 字符串**

1. **以下关于函数的描述,错误的是（ ）。**

**A. 函数是一种功能抽象**

**B. 使用函数的目的只是为了增加代码复用**

**C. 函数名可以是任何有效的Python标识符**

**D. 使用函数后,代码的维护难度降低了**

1. **以下的函数定义中,错误的是（ ）。**

**A. def vfunc(s,a=1,\*b):**

**B. def vfunc(a,3):**

**C. def vfunc(a,\*\*b):**

**D. def vfunc(a,b=2):**

1. **下面代码的输出结果是（ ）。**

**dict\_data = {"化1704":33, "化1702":28, "化1701":34, "化1703":30}**

**print(sorted(dict\_data.items(),key=lambda x:x[1]%10))**

**A.['化1701', '化1702', '化1703', '化1704']**

**B.[('化1701', 34), ('化1702', 28), ('化1703', 30), ('化1704', 33)]**

**C.[('化1703', 30), ('化1704', 33), ('化1701', 34), ('化1702', 28)]**

**D.[('化1702', 28), ('化1703', 30), ('化1704', 33), ('化1701', 34)]**

1. **以下程序的输出结果是（ ）。**

**def calu(x=2,y=4):**

**return(x\*y)**

**a='20'**

**print(calu(a),end=',')**

**A. 20202020, B. 8, C. 20202020,8 D. 24,**

1. **下面代码的输出结果是（ ）。**

**def hub(ss, x = 2.0,y = 4.0):  
      ss += x \* y  
ss = 10  
print(ss, hub(ss, 3))**

**A. 22.0 None B. 10 None C.22 None D. 10.0 22.0**

1. **关于函数的可变参数，可变参数＊\*args 传入函数时存储的类型是（ ）。**

**A. tuple B. list C. set D. dict**

1. **以下程序的输出结果是( )。**

**def Hello(name, age=60):**

**if age>50:**

**print("您好！"+ name + "大伯")**

**elif age>30:**

**print("您好！"+ name + "大哥")**

**else:**

**print("您好！小" + name)**

**Hello(age=29, name="张")**

**A．您好！张大伯**

**B．您好！张大哥**

**C．您好！小张**

**D．函数调用出错**

1. **以下选项中,不是Python对文件的打开模式的是（ ）。**

**A. 'w' B. 'r' C. 'a+' D. 'c'**

1. **在Python语言中，使用open()打开一个Windows操作系统D盘下的文件,以下选项中对路径的表示错误的是（ ）。**

**A. D:\PythonTest\a.txt**

**B. D://PythonTest//a.txt**

**C. D:\\PythonTest\\a.txt**

**D. D:/PythonTest/a.txt**

1. **下面代码的输出结果是（ ）。**

**try:**

**tp=(1,2,3)**

**tp[1]=1**

**except Exception:**

**print("计算错误")**

**else：**

**print("1")**

**A. 0 B. 计算错误 C. 1 D. 没有输出**

1. **关于以下代码的描述,错误的选项是（ ）。**

**with open('abc.txt','r+') as f:  
 lines = f.readlines()  
 for item in lines:  
 print(item)**

**A. 执行代码后,abc.txt文件未关闭,必须通过close()函数关闭**

**B.打印输出abc.txt文件内容**

**C. item是字符串类型**

**D. lines是列表类型**

1. ****若字符串s= 'a\nb\\cd'，则len(s)的值是（ ）。****

**A.5 B. 6 C.7 D. 8**

**二、填空题（按空填写，共20空，每空2分，共40分）**

1. **若x=0,y=1,z=3,则表达式x or y and z的值为： ( 1 )**
2. **已知列表对象x = ['11', '2', '3']，则表达式 max(x) 的值为 ( 2 ) 。**
3. **在函数内部可以通过关键字 ( 3 ) 来定义全局变量。**
4. **Python表达式5//3-True\*3+False的值为 ( 4 ) 。**
5. **表达式 sorted([111, 2, 33], key=lambda x: len(str(x))) 的值为 ( 5 ) 。**
6. **表达式 chr(ord('A')+2) 的值为 ( 6 ) 。**
7. **已知函数定义 def func(\*p):return sum(p)，那么表达式 func(1,2,3, 4) 的值为 (7) 。**
8. **已知 x = [1,2,3,4,5]，那么执行语句 x[::2] = range(3) 之后，x的值为 ( 8 ) 。**
9. **执行下面代码后，程序运行结果是 ( 9 ) 。**

**def fun(m,n):**

**if m>=n:**

**print(m,end="")**

**else:**

**fun(m+1,n)**

**print(m,end="")**

**fun(1,2)**

1. **表达式 {1, 2, 3, 4} - {3, 4, 5, 6}的值为 ( 10 ) 。**
2. **假设已有列表a=[9，6，7，10],且b=a,则执行语句b[2]=0后，则a的值为： ( 11 )。**
3. **执行下列Python语句后的输出结果为： ( 12 ) 。**

**i=-1**

**while i<0: i\*=i**

**print(i)**

1. **下列Python语句的输出结果是 ( 13 ) 。**

**s=['a','b']**

**s.append([1,2])**

**s.extend([5,6])**

**s.insert(10,8)**

**s.pop()**

**s.remove('b')**

**s[3:]=[]**

**s.reverse()**

1. **执行下面代码后，变量c的值为 ( 14 ) 。**

**def func(a,b):**

**c=a\*\*2+b**

**a=b**

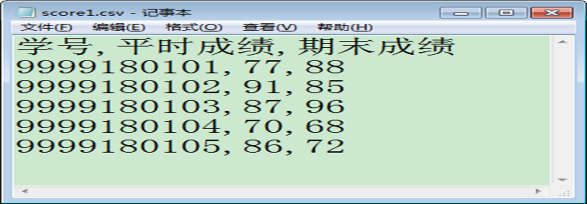
**return c**

**a=10**

**b=100**

**c=func(a,b)+a**

1. **有score1.csv文件如下，计算学生总评成绩（总评成绩=平时成绩\*0.4+期末成绩\*0.6），并按总评成绩降序排列后保存到一个新文件。请完善程序。**

****

**(15)**

**with open('score1.csv', "r", encoding='utf-8') as file:**

**data = list(csv.reader(file))[1:] # 去除标题行**

**print(data)**

**for d in data:**

**d.append(str(round(int(d[1]) \* 0.4 + int(d[2]) \* 0.6)))**

**print(data)**

**data.sort(key= (16) , reverse=True)**

**data.insert(0, ['学号', '平时成绩', '期末成绩', '总评成绩'])**

**with open('score\_total.csv', "w", newline='') as file:**

**writer = csv.writer(file)**

**writer.writerows(data)**

1. **以下程序将题目中给出的文本进行分词，并将准确而不冗余的分词结果以列表形式输出，如输出结果为:['2021', '年', '7', '月', '1', '日', '是', '中国共产党', '成立', '100', '周年', '纪念日']，请完善程序。**

**import ( 17 )**

**s="2021年7月1日是[中国共产党](https://baike.baidu.com/item/%E4%B8%AD%E5%9B%BD%E5%85%B1%E4%BA%A7%E5%85%9A/117227" \t "https://baike.baidu.com/item/%E5%BA%86%E7%A5%9D%E4%B8%AD%E5%9B%BD%E5%85%B1%E4%BA%A7%E5%85%9A%E6%88%90%E7%AB%8B100%E5%91%A8%E5%B9%B4/_blank)成立100周年纪念日"**

**ls= ( 18 )**

**print(ls)**

1. **以下程序的年度和录取率基本信息如下表，计算七年平均录取率，请完善程序，每个空格填写一个表达式。**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **2006** | **2007** | **2008** | **2009** | **2010** | **2011** | **2012** |
| **57%** | **56%** | **57%** | **62%** | **69%** | **72%** | **75%** |

**rate\_years=[(2006,57),(2007,56),(2008,57),(2009,62),(2010,69),(2011,72),(2012,75)]**

**rates=[ ( 19 ) for item in rate\_years] #录取率列表**

**avg= ( 20 )**

**print("七年平均录取率为{:.2f}%".format(avg))**