Python 习题 9 参考答案

单选题

DDCD

填空题

1.输入一个自然数 n，随机生成 n 个[50,100]间的数，并保存在 1.txt 文件中，每个数独占一行。

```
from random import *         #导入随机函数
n=int(input())               #输入自然数 n
f=open("1.txt","w")          #以写方式打开文件
for i in range(n):           #指定写入 n 个数
    x=randint(50,100)        #生成一个随机数
    f.write(str(x)+'\n')     #将随机数 x 转成字符串和换行符一起写入文件
f.close()                    #关闭文件，存储数据
```

2.分三行将学号、姓名、和班级写入文本文件 name.txt 中，然后将文件 name.txt 中的内容输出。

```
fo = open('name.txt','w+')
ls = ['2000','姚期智','计算机 1 班']
for line in ls:
    fo.write(line+'\n')
fo.seek(0)
for line in fo:
    print(line,end="")
fo.close()
```

3. 在读写文件的过程中，tell()方法可以获取当前的读写位置。

4. seek 方法用于移动指针到指定位置，该方法中 offset 参数表示要偏移的字节数。

5. 统计文本文件中字数，每个字母按一个汉字计算，不包括所有标点符号和空格

```
name=input()
fi=open(name,"r",encoding="UTF-8")
txt=fi.read()
for c in "'，。？！ @#¥%……&*；（:; ）--+|、•： """'\n'":
    txt=txt.replace(c,"")
print(len(txt))
fi.close()
```

6. 编写一个程序，提示用户输入一个文件名然后统计在不计大小写的情况下每个字符的出现次数。

```
def main():
    filename = input("Enter a filename: ").strip()
    infile = open(filename, "r")          # Open the file
    counts = 26 * [0]                     # Create and initialize counts
    for line in infile:                   # Invoke the countLetters function to count each letter
        countLetters(line.lower(), counts)
    ''' Display results'''
    for i in range(len(counts)):
        if counts[i] != 0:
            print(chr(ord('a') + i) + " appears " + str(counts[i])
              + (" time" if counts[i] == 1 else " times"))
    infile.close()                        # Close file

'''Count each letter in the string'''
def countLetters(line, counts):
    for ch in line:
        if ch.isalpha():
            counts[ord(ch) – ord(‘a’)] += 1

main()                                    # Call the main function
```

编程题

1. data.txt 中保存若干行文本。请编写一个程序读取文件中文本，并统计输出文本的有效行数，然后将结果保存到 result.txt 中。

```
fp1=open("data.txt","r",encoding="utf-8")
n=0
while(1):
    x=fp1.readline()
    if x!="":
        if x.strip()!="": n+=1
    else:
        break
fp2=open("result.txt","w")
fp2.write("有效行数为:"+str(n)+"行")
fp2.close()
```

2. data.txt 中保存有 n 个单词，每个单词一行。请编写一个程序从文件中将单词读出，找到最长的单词，然后将其保存到 result.txt 中。程序须保存 test.py 中

```
fp1=open("data.txt","r",encoding="UTF-8")
s=fp1.readlines()
maxlen=0
word=[]
for i in s:
    if len(i.strip())>maxlen: maxlen=len(i.strip())
for i in s:
    if len(i.strip())==maxlen: word.append(i.strip())
fp2=open("result.txt","w")
if len(word)==1:
    #print(f"The longest word is: {word[0]}")
    fp2.write("The longest word is: " + word[0])
else:
    # print(f"The longest words are: {','.join(word)}")
    fp2.write(f"The longest words are: {','.join(word)}")
```

3. read the contents of the sample.txt file, analyze the character distribution of the lowercase letters a-z, count the total number of characters in the file, and output the result in alphabetical order to result.txt file.

```
fp1=open("sample.txt","r",encoding="utf-8")
s=fp1.read()
d={}
sum=0
for i in s:
    if i.isalpha():
        d[i.lower()]=d.get(i.lower(),0)+1
        sum+=1
fp2=open("result.txt","w")
print(f"There are a total of {sum} characters, where each letter is distributed as follows:")
fp2.write("There are a total of "+str(sum)+" characters, where each letter is distributed as follows:\n")
d1=list(d.items())
d1.sort()
for k,v in d1[:-1]:
    if v!=0: print(f"{k}:{v}, ",end="")
    if v!=0: fp2.write(str(k)+":"+str(v)+", ")
if d1[-1][1]!=0: print(f"{d1[-1][0]}:{d1[-1][1]}")
if d1[-1][1]!=0: fp2.write(str(d1[-1][0])+":"+str(d1[-1][1]))
```