

# Problem Set 4 实验报告

17340027 姚洁倩

## A.

### 1.题目描述：

要求使用递归求出一个字符串的全排列，并且结果不包含重复的排列。

### 2.解题思路

文档中给出了求这些全排列的方法：若字符串长度大于一，则递归求它除去第一个字符的字符串的所有全排列，然后将第一个字符插入在字符串的全排列的所有可能位置即可得到此字符串的全排列；若字符串长度等于一，为递归终止的条件，此时直接返回这个字符即可。

另外，在求全排列的时候要注意有没有重复的可能，在求得一个字符串的一个排列的时候，要检查这个排列是否已经在存储全排列的列表中出现过。若已经出现过，则不再将其添加进列表，这样便可以避免重复的可能。

### 3.部分测例及运行截图

```
PS C:\coding\python\p4-17340027-姚洁倩> python .\ps4a.py.  
Input: abc  
Expected Output: ['abc', 'acb', 'bac', 'bca', 'cab', 'cba']  
Actual Output: ['abc', 'bac', 'bca', 'acb', 'cab', 'cba']  
  
Input:  
Expected Output: []  
Actual Output: []  
  
Input: aaa  
Expected Output: ['aaa']  
Actual Output: ['aaa']  
  
Input: bbc  
Expected Output: ['bbc', 'bcb', 'cbb']  
Actual Output: ['bbc', 'bcb', 'cbb']
```

测试了默认给出的测例，成功通过。

测试了空字符串，输出也没有问题。

测试了包含相同字母的字符串，用来检测全排列是否成功地排除了重复的排列。运行结果显示，代码成功地排除了重复的排列。

## B.

### 1.题目描述

题目要求对明文进行凯撒加密，并且在未知密钥的情况下能对密文进行解密，找出最有可能是明文的情况，输出对应密钥以及解密后的文本。要使用类来实现。

### 2.解题思路

(1) 先弄清楚各个类的功能：

- ◆ Message 为基类，类有两个公有成员变量：message\_text 和 valid\_words，另外这个类有两个比较重要的函数：build\_shift\_dict——根据参数 shift 构造从明文字母到密文字母的一一映射关系，apply\_shift——根据构造好的映射关系，将 message\_text 进行转换。
- ◆ PlaintextMessage 类继承自 Message 类，除了 Message 中原有的成员变量以外，还有 shift、encryption\_dict 和 message\_text\_encrypted 这三个成员变量，分别指示明文字母和密文字母之间的间隔、明文字母和密文字母之间的一一映射关系、加密后的密文内容。此外，还有 change\_shift 函数，能够根据 shift 的改变能够相应的改变映射关系以及加密后的密文。
- ◆ CiphertextMessage 类继承自 Message 类，成员变量即 Message 类中的两个。另外还有一个函数 decrypt\_message：将 shift 从 0-25 一个个尝试，对 message\_text 应用变换进行解密，对变换后的文本检查有意义的单词个数，选取有最大有意义单词数的文本作为解密后所得的明文。

(2) 实现类中函数的功能

- ◆ 几个 get 函数直接返回相对应的变量即可
- ◆ Build\_shift：使用如下语句对字母进行转换

```
shift_dic = {}
for lower in string.ascii_lowercase:
    shift_dic[lower] = chr((ord(lower)-ord('a')+shift)%26+ord('a'))
for upper in string.ascii_uppercase:
    shift_dic[upper] = chr((ord(upper)-ord('A')+shift)%26+ord('A'))
return shift_dic
```

使用字母的 ascii 码值减去大写字母 A 或小写字母 a 的 ascii 码值，再加上 shift 的值，得到的结果模 26，最后加上原先减去的 A 的 ascii 码值，再转化为字符即是所要变换成为的密文字母。

- ◆ apply\_shift：遍历 message\_text 里面的每一个字符，查看此字符是否在字典中有映射的关系，若有则转化成对应的字母再加入密文；若没有，则说明它是空格或者标点符号一类的，不需要进行转化，在密文中直接加入它本身即可。
- ◆ change\_shift：修改类中 shift 这个成员变量，重新求一次映射的字典并更新，最后，使用新的字典对文本再次进行转换。
- ◆ decrypt\_message：对 message\_text 尝试进行解密，方法是使用 for 循环，将 shift 的值从 0 依次递增到 25，每次都使用当前的 shift 值对文本进行转换，使用 apply\_shift 函数，并且查看转换后的文本中包含的有意义的单词数，选取单词数最多的作为结果返回。

### 3.部分测例及运行截图

```

PS C:\coding\python\p4-17340027-姚洁倩> .\ps4b.py
Loading word list from file...
55901 words loaded.
Expected Output: jgnnq
Actual Output: jgnnq
Loading word list from file...
55901 words loaded.
Expected Output: (24, 'hello')
Actual Output: (24, 'hello')

Loading word list from file...
55901 words loaded.
Expected Output: Qtsits gwnilj nx kfqqnsl itbs
Actual Output: Qtsits gwnilj nx kfqqnsl itbs
Loading word list from file...
55901 words loaded.
Expected Output: (21, 'London bride is falling down')
Actual Output: (21, 'London bridge is falling down')

Loading word list from file...
55901 words loaded.
Expected Output: Dynki'c qyyn!
Actual Output: Dynki'c qyyn!
Loading word list from file...
55901 words loaded.
Expected Output: (16, "Today's good!")
Actual Output: (16, "Today's good!")

Original text: Xoxy Tzcfsm wg o amhvwqoz qvofqhsf qfsohsr cb hvs gdif ct o acasbh hc vszd qcjs
f ob wbgittwqwsbhz dzobbsr voqy. Vs vog pssb fsuwghsfsr tcf qzogsg oh AWH hkwqs pstcfs, pih v
og fsdcfhsrzm bsjsf doggsr oqzog. Wh vog pssb hvs hforwhwb ct hvs fsgwrsbhg ct Sogh Qoadig hc
psqcas Xoxy Tzcfsm tcf o tsk bwuvhg soqv msof hc sriqohs wbqcaubu ghirsbhg wb hvs komg, asobg,
obr shvwqg ct voqywb.
Loading word list from file...
55901 words loaded.
The decryption of the story:
(12, 'Jack Florey is a mythical character created on the spur of a moment to help cover an insu
fficiently planned hack. He has been registered for classes at MIT twice before, but has report
edly never passed aclass. It has been the tradition of the residents of East Campus to become J
ack Florey for a few nights each year to educate incoming students in the ways, means, and ethi
cs of hacking.')
```

测试了原有的样例，加密和解密均成功通过，解密出来的文字和原文匹配百分之百  
测试了带有空格的句子，加密和解密均成功通过，解密出来的文字和原文匹配百分之百  
测试了加上标点符号的句子，加密和解密均成功通过，解密出来的文字和原文匹配百分之百  
最后，对 story 进行解密，找出了一个最可能的版本，阅读后发现，此文本是有意义的，故  
可认为解密成功。

## C.

### 1. 题目描述：

题目要求实现对文本的元音字母进行替换加密，并且在不知道密钥的情况下尝试对密文进行解密，找出最可能的明文。

### 2. 解题思路

(1) 先弄清楚各个类的功能

- ◆ SubMessage 为基类，有 message\_text 和 valid\_words 两个成员变量，另外有 build\_transpose\_dict 和 apply\_transpose 两个函数，分别用于建立明文和密文之间的一一对应关系以及根据这个建立的字典对 text 进行转换。

- ◆ EncryptedSubMessage 继承自 SubMessage, 成员变量即是 SubMessage 中的两个。此外还有一个 decrypt\_message 函数, 对 text 进行解密。

(2) 对类中的函数进行实现

- ◆ Build\_transpose\_dict: 建立从明文字母到密文字母的一一对应映射关系, 遍历小写和大写字母, 若为元音则映射至对应的转换后的字母, 若为辅音, 则映射为它本身。
- ◆ Apply\_transpose: 对文本进行应用变换, 遍历文本中的每一个字母, 将所有的元音字母映射成它相对应的转换即可得到转换之后的文本。
- ◆ Decrypt\_message: 解密信息。先求五个元音字母的全排列, 尝试每一种排列所指示的转换, 将文本进行解密, 统计解密所得的文本的有意义单词数, 选取最多的那个作为所得的密文。

### 3.部分测例及运行截图

```
PS C:\coding\Python\p4-17340027-姚洁倩> python .\ps4c.py
Loading word list from file...
  55901 words loaded.
Original message: Hello World! Permutation: eaiuo
Expected encryption: Hallu Wurld!
Actual encryption: Hallu Wurld!
Loading word list from file...
  55901 words loaded.
Decrypted message: Hello World!
Loading word list from file...
  55901 words loaded.
Original message: It's a good idea. Permutation: iouea
Expected encryption: Ut's i geed udoi.
Actual encryption: Ut's i geed udoi.
Loading word list from file...
  55901 words loaded.
Decrypted message: It's a good idea.
Loading word list from file...
  55901 words loaded.
Original message: Bravo! This code is splendid Permutation: oueai
Expected encryption: Brova! Thes cadu es splunded
Actual encryption: Brova! Thes cadu es splunded
Loading word list from file...
  55901 words loaded.
Decrypted message: Bravo! This code is splendid
```

测试了所给的样例, 成功通过。

测试了两个带有空格和标点符号的样例, 成功通过。