**一、实验课程编码：106006**

**二、实验课程名称：媒体内容安全**

**三、实验项目名称：**基于Python3的古典密码编程

**四、实验目的**

1． 置换密码的设计和实现。

2. 栅栏密码的设计与实现。

**五、主要设备**

安装Windows和python软件的个人计算机

**六、实验内容**

2．凯撒密码的设计和实现

2.1

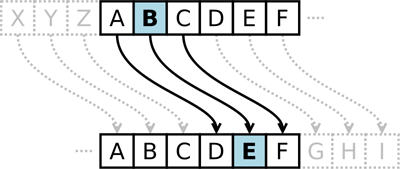
凯撒密码是罗马扩张时期朱利斯•凯撒（Julius Caesar ）创造的，用于加密通过信使传递的作战命令。它将字母表中的字母移动一定位置而实现加密。例如如果向右移动2 位，则字母A 将变为C ，字母B 将变为D ，… ，字母X 变成Z ，字母Y 则变为A ，字母Z 变为B 。因此，假如有个明文字符串“Hello ”用这种方法加密的话，将变为密文：“Jgnnq ”。而如果要解密，则只要将字母向相反方向移动同样位数即可。如密文“Jgnnq ”每个字母左移两位变为“Hello ”。这里，移动的位数“2 ”是加密和解密所用的密钥。

该密码以被古罗马皇帝凯撒使用而闻名，用于和将军们进行联系。它通常被作为其他更复杂的加密方法中的一个步骤，例如维吉尼亚密码。但是和所有的利用字母表进行替换的加密技术一样，恺撒密码非常容易被破解，而且在实际应用中也无法保证通信安全。

其基本思想是：通过把字母移动一定的位数来实现加密和解密。例如，密匙是把明文字母的位数向后移动三位，那么明文字母B就变成了密文的E,依次类推，X将变成A，Y变成B，Z变成C。

该密码以被古罗马皇帝凯撒使用而闻名，用于和将军们进行联系。它通常被作为其他更复杂的加密方法中的一个步骤，例如维吉尼亚密码。但是和所有的利用字母表进行替换的加密技术一样，恺撒密码非常容易被破解，而且在实际应用中也无法保证通信安全。

其基本思想是：通过把字母移动一定的位数来实现加密和解密。例如，密匙是把明文字母的位数向后移动三位，那么明文字母B就变成了密文的E,依次类推，X将变成A，Y变成B，Z变成C。



因此，可使用VC++实现对输入明文的凯撒密码解密

2.2 凯撒密码的推广

凯撒密码的密钥空间有限，导致安全性极差，那么如何对它进行改进呢？

一个简单的思路是：将明文字符进行任意替换，也就是“a”可以替换为26个字符，“b”在剩余25个字符中任意选择一个，依次类推。我们将这样的凯撒密码称之为“凯撒密码推广算法”。

那么“凯撒密码推广算法”的密钥量是多少呢？

答案是26！= 4.0329146×10^{26} （26的阶乘）。大家可以用Python来计算下26的阶乘。这是一个极为巨大的数字，比DES的密钥大10个数量级。试想下，如果计算机每秒运行1万次穷举攻击算法，那么完全运算完需要多长的时间呢？

class ECaesarCipher( ):

def \_\_init\_\_(self, message=None):

self.message = message

self.cipherMessage=""

#获取“a”到”z“的顺序字符表

pT=tuple(string.ascii\_lowercase)

#建立随机化的密文字符表

cT=[]

temp=range(0,26)

random.shuffle(temp)

for x in temp:

cT.append(pT[x])

#建立密码本，即明文、密文对照表，这里使用了字典结构

self.keyTable={}

for x,y in zip(pT,cT):

self.keyTable[x]=y

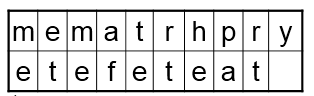
算法建立ECaesarCipher类，支持大小写英文字母、空格作为明文字符，初始化时随机产生明密文对照表。这个明密文对照表也可以称为密码本，是必须要私密保存的密钥。

3．栅栏密码的设计与实现

该密码的思想是“按列写入，按行读出”，就是把要加密的明文分成N个一组，然后把每组的第1个字连起来，形成一段无规律的话。

例如：用深度为2的栅栏技术加密明文“meet me after the party”

可写为



**七、实验步骤**

1．编写置换密码加解密算法，可以对相关英文文本进行加解密。

2．编写栅栏密码加解密算法，可以对相关英文文本进行加解密。

**八、实验结果**

总结置换密码和栅栏密码的转换公式及编程实现的算法并写成实验报告。实验报告以电子版形式撰写，程序代码作为报告的附件，实验后第二次实验室课时提交。