**Python程序设计**

**实验报告**

**实验题目：凯撒密码的设计**

**专 业：人工智能**

**姓 名：洪伟斌**

**学 号：22920192203959**

**实验日期**：3.25

**一. 实验目的**

1.熟悉python语言设计程序的方法

2.了解凯撒密码是什么

3.分别设计凯撒密码的加密程序及解密程序

1. **实验内容**

使用python语言设计一个中文的动态的凯撒密码。我对动态这个词的理解是每次输入同一个明文时，程序能给出不同的加密后的密文。

1. **实验步骤以及结果**

以下是我设计的程序

1. 加密程序

import random  
a=input("请输入明文：")  
b=''  
l=len(a)  
ran=random.randint(0,1500)  
for i in range(l):  
 b+=str(chr(ord(a[l-i-1])+(l+3)%7+i\*2+ran%30)) #具体操作  
b=b+chr(ran+l%3+20000)  
print("加密后的密文为："+b)

如图所示，先将代加密的密码输入，输入后用ord取得输入的每个汉字的unicode的值，然后将这个值进行右移动(具体如何移动请看代码)，然后再用chr函数将每个移动后的unicode的值转换成中文，最后加到字符串中。这里还对字符串进行了转置，即将第一个字符加密后的密文对应的是最后一位，第二位对应的则是倒数第二位，以此类推。

由于要让每次加密都不一样，在这里引入了一个随机数，并通过处理将这个随机数也转换成了相应的中文，并加在字符串的最后一位，因此解密出来的密文个数会比明文多一个。因为字符串数值在左右移动时，其左右移动的距离与随机数有关，因而能够做到每次加密过后的密文都不一样。

1. 解密程序

a=input("请输入密文：")  
b=''  
l=len(a)  
ran=(ord(a[l-1])-20000)-(l-1)%3  
for i in range(l-1):  
 b+=str(chr(ord(a[i])-(l+2)%7-i\*2-ran%30))  
c=b[::-1] #转置  
print("解密后的明文为："+c)

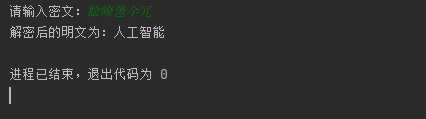
解密程序相对来说就是对加密程序中字符数值的移动进行了逆操作，由于加密过程中含有一个随机数，因此要从密文中的最后一个字符先解出相应的随机数，紧接着进行逆操作即可，由于这边我将明文和相应的密文转置了，因此解密过后还要将解密出的明文进行一次转置，如图所示。

1. 运行结果

①加密：



解密：



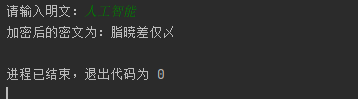
②加密：



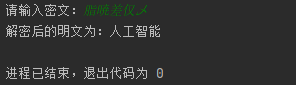
解密：



③加密：



解密：



上面三个例子输入的明文虽然一样，但加密后的密文都不一样，但是还是能够用解密程序解出政权的明文。

**四．实验总结**

通过本次实验，我了解了凯撒密码是什么，还学会了python中一些函数的用法，比如ord和随机数的取法，除此之外使用python对字符串的处理也更加得心应手。

本次实验的重点是动态的问题，我只能想到通过增加一个字符来达到“动态”的效果，如果有能够不增加字符还能使其达到“动态”的话那要比这个要好很多，这也引发了我的思考。