**人工智能程序设计**

**实验报告**

**实验题目：设计一个凯撒密码加密和解密程序**

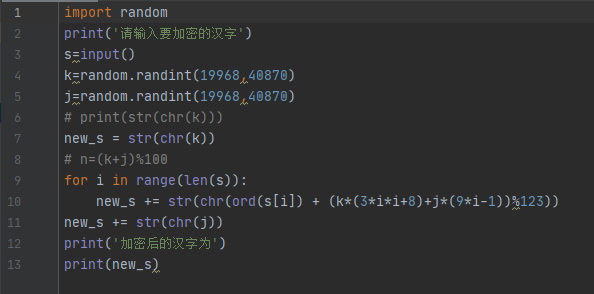
**专 业：人工智能**

**姓 名：陈億龙**

**学 号：22920192203918**

**实验日期**：**2021/3/25**

1. 完整代码
2. **加密程序**



算法思想：（1）首先输入需要加密的汉字

（2）构造两个在19968到40870之间的随机数，用于确定步长

（3）将上一步中得到的两个随机数通过chr（）函数返回代表指定unicode的字符，

并进一步用str（）函数转化为字符串类型，便于操作.

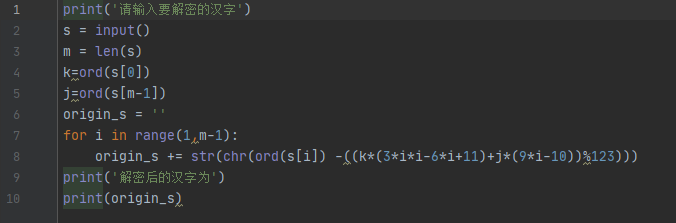
（4）对输入的汉字从头到尾进行for循环，通过公式 (k\*(3\*i\*i+8)+j\*(9\*i-1))%123 求出每一个输入的汉字所对应的步长。

（5）通过ord（）函数将每一个输入的汉字返回为unicode，并加上上一步中求出的各自的步长，得到新的unicode，

再通过chr（）函数和str（）函数，得到新的字符串。

（6）将随机数k对应的汉字放于字符串new—s的开头处，随机数j对应的汉字放于其结尾处，便于进一步的解码操作，上一步中得到的新的字符串则放于中间，从而得到完整的字符串new—s，即为加密后的汉字。

1. **解码程序**



算法思想：（1）首先输入待解密的汉字。

（2）通过ord（）函数将第一个汉字以及最后一个汉字返回为unicode，并分别存储在k、j中。

（3）创建一个空的字符串origin—s，用于存放解密后的汉字。

（4）对输入的汉字从正数第二个到倒数第二个进行for循环，通过公式(k\*(3\*i\*i-6\*i+11)+j\*(9\*i-10))%123

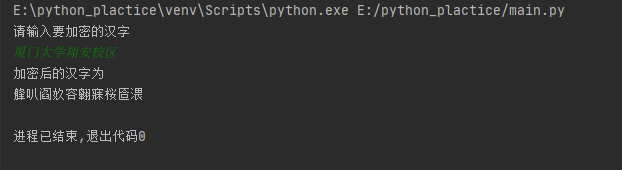
求出每个汉字在被加密时所增加的步长。并用ord（）函数得出加密后每个汉字对应的unicode，后者减去前者，

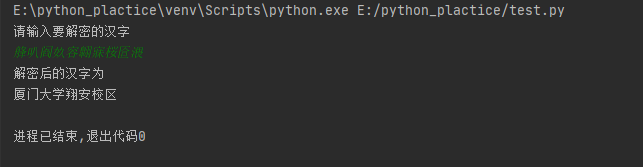
即为解密后的汉字的unicode，再用chr（）函数及str（）函数将其转化为字符串。

（5）将每一步循环的字符串连接，得到完整的字符串origin—s，即为解密后的汉字。

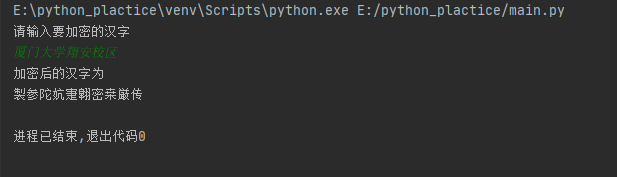
**3、运行截图**

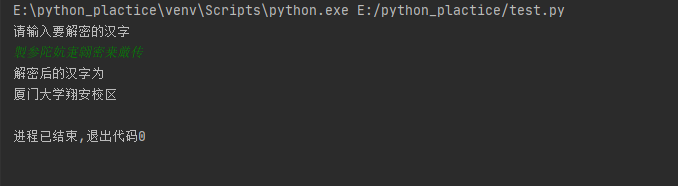
第一组：



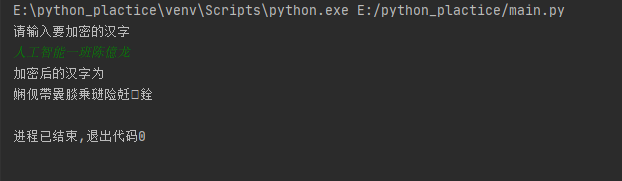


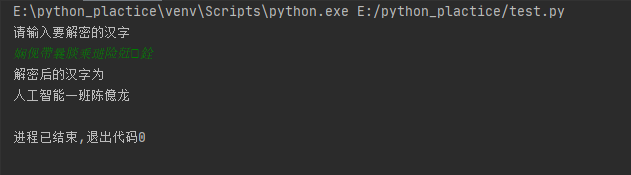
第二组：





第三组：





1. **实验优缺点分析**
2. **优点**
3. 该程序的加密的步长随着每次random出的随机数的不同而不同，且每个汉字加密的步长也互不相同，即使两次完全相同的输入也能得到不同的加密汉字，故具有一定的随机性。
4. 计算步长的加密公式为二次非线性方程，且还要对123取模，具有一定的复杂度。
5. **缺点**
6. 该程序需要额外添加两个汉字用于计算步长以及之后的破解工作，但经过大数据统计之后，可以发现这两个汉字为随机汉字，故会露出一定破绽。
7. **实验总结**

**本次实验我一共做了四个版本，从最初的所有汉字加密步长均相同且不随机，到加密步长相同且随机，再到加密步长各不相同但呈线性分布，直到将线性改善为非线性而完成最终的版本，**

**我尝试了各种方法来实现算法的正确性随机性复杂性，在这个过程中，我学到的不仅是编程的技巧，更多的是思维的灵活、逻辑的缜密，而作为初学者，我也体会到算法是需要不断完善改进的，**

**当我回顾我第一个版本的代码时，发现它有很多的漏洞以及缺陷，但经过实验课上自己的思考以及老师们的帮助，最终完成了一个相对可行的代码，我也体会到了其中的乐趣。**