测试结果

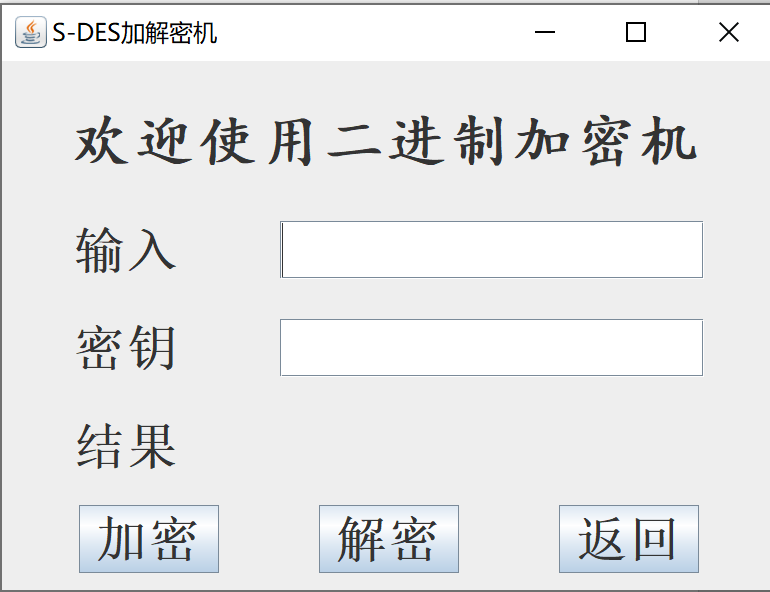
根据上述5个关卡的测试要求，分别提供相应的测试结果，灵活使用文字、表格、截图的方式来展现。

**3.1 第1关：基本测试**

根据S-DES算法编写和调试程序，提供GUI解密支持用户交互。输入可以是8bit的数据和10bit的密钥，输出是8bit的密文。



选择二进制加密



输入明文10101010，密文1111100000



**现对生成的密文（00111011），使用相同的密钥（1111100000）进行解密**



**加解密结果相匹配。**

**3.2 第2关：交叉测试**

考虑到是**算法标准**，所有人在编写程序的时候需要使用相同算法流程和转换单元(P-Box、S-Box等)，以保证算法和程序在异构的系统或平台上都可以正常运行。

设有A和B两组位同学(选择相同的密钥K)；则A、B组同学编写的程序对明文P进行加密得到相同的密文C；或者B组同学接收到A组程序加密的密文C，使用B组程序进行解密可得到与A相同的P。

第一组同学使用密钥1111100000对明文10101010进行加密，得到密文00111011



第二组同学对上一位同学同样对上述明文同相同密钥加密

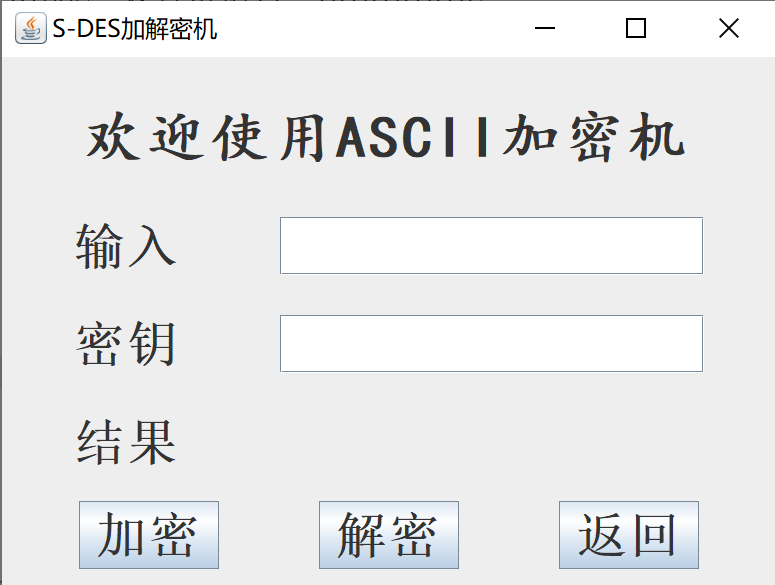
可见结果相同。

**3.3 第3关：扩展功能**

考虑到向实用性扩展，加密算法的数据输入可以是ASII编码字符串(分组为1 Byte)，对应地输出也可以是ACII字符串(很可能是乱码)。



选择ASCII加密



输入字符串“ASCii”设置密钥为：1100100111



**现对生成的密文（±®`££），使用相同的密钥（1100100111）进行解密**



**加解密结果相匹配**

**3.4 第4关：暴力破解**

假设你找到了使用相同密钥的明、密文对(一个或多个)，请尝试使用暴力破解的方法找到正确的密钥Key。在编写程序时，你也可以考虑使用多线程的方式提升破解的效率。请设定时间戳，用视频或动图展示你在多长时间内完成了暴力破解。

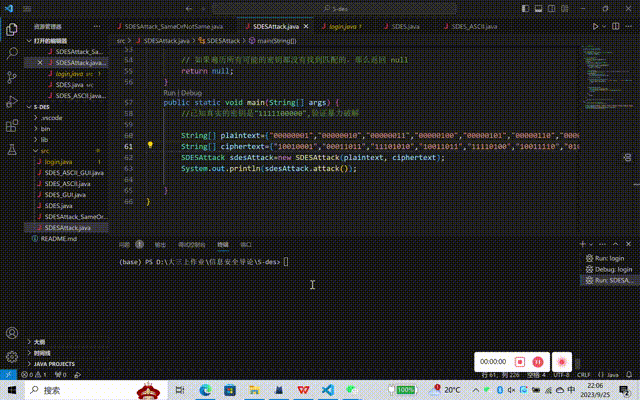
已知的19个明文，其真实的密钥为：1111100000：

{"00000001","00000010","00000011","00000100","00000101","00000110","00000111","00001000","00001001","00001010","00001011","00001100","00001101","00001111","00010000","00010001","00010010","00010011","00010100"}

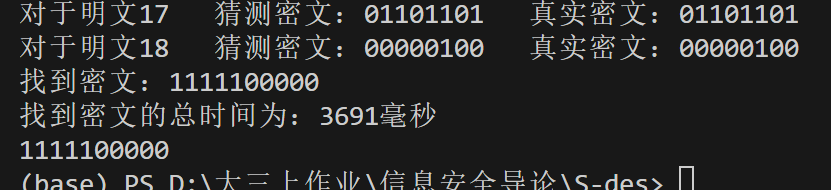
与这19个明文所对应的密文：

{"10010001","00011011","11101010","10011011","11110100","10011110","01001011","01100100","11101011","00110101","11100000","00100001","01001010","10100101","01000001","01000110","00110000","01101101","00000100"}

将其输入到暴力破解算法中，验证找出的密钥是否和真实的密钥相同：（以下为gif）



为防止看不清楚，下面是结果截图：

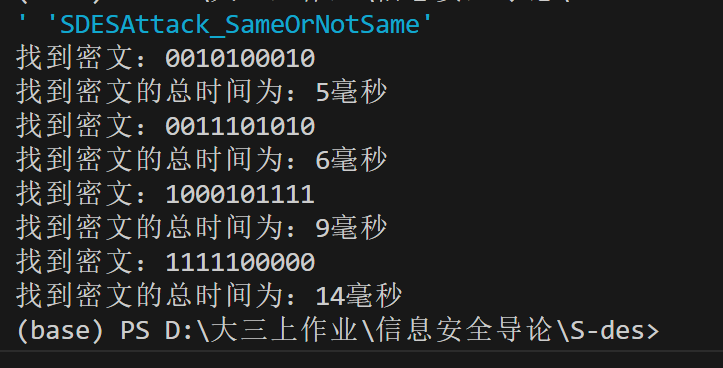


由结果可知，暴力破解出来的密钥1111100000和原来真实的密钥是相同的。

整个暴力破解程序的运行时间为3691毫秒，但是其中存在这程序设计的时间复杂度问题，可能与真实值之间有差异。但是无论如何我们都可以发现S-DES加密被破解的时间是非常短的。说明S-DES的安全性并不高。

**3.5 第5关：封闭测试**

根据第4关的结果，进一步分析，对于你随机选择的一个明密文对，是不是有不止一个密钥Key？进一步扩展，对应明文空间任意给定的明文分组P\_{n}，是否会出现选择不同的密钥K\_{i}\ne K\_{j}加密得到相同密文C\_n的情况？

1、当随机选取一个明密文对：明文（00000001），密文（10010001）对此采用第四关的暴力破解方式进行破解结果如下：  


发现四个密钥对此明文进行加密获得的密文是相同的。

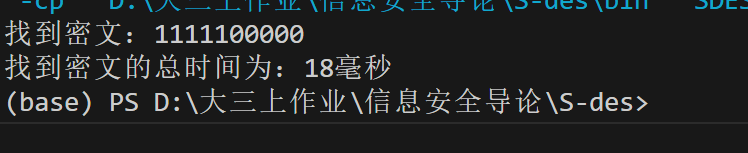
2、对于明文分组和密文分组：

明文：

{"00000001","00000010"};

密文：

{"10010001","01001011"};

使用暴力破解结果：  


只找到一个密钥。

后续经过多次实验，发现明文空间任意给定的明文分组没有找到出现选择不同密钥加密得到相同的密文的情况。