S-DES 加密算法工具 用户指南

一、程序简介

本程序是基于 Python 实现的 S-DES（Simplified Data Encryption Standard）加密算法学习与演示工具。通过图形化界面，用户可以方便地进行二进制加密解密、文本加解密、密钥暴力破解及密钥冲突分析等操作。程序适用于密码学课程教学、算法演示及实验分析场景。

二、运行环境与启动方式

1. Python 版本：Python 3.10
2. 依赖库：tkinter、threading、time、itertools（均为标准库）
3. 启动方式：在命令行中运行 python DES_1.py，程序启动后，将显示带有多个功能选项卡的图形化界面。

三、界面结构与主要功能

程序界面主要分为三个标签页：

1. 二进制加解密；2. 文本加解密；3. 密钥分析

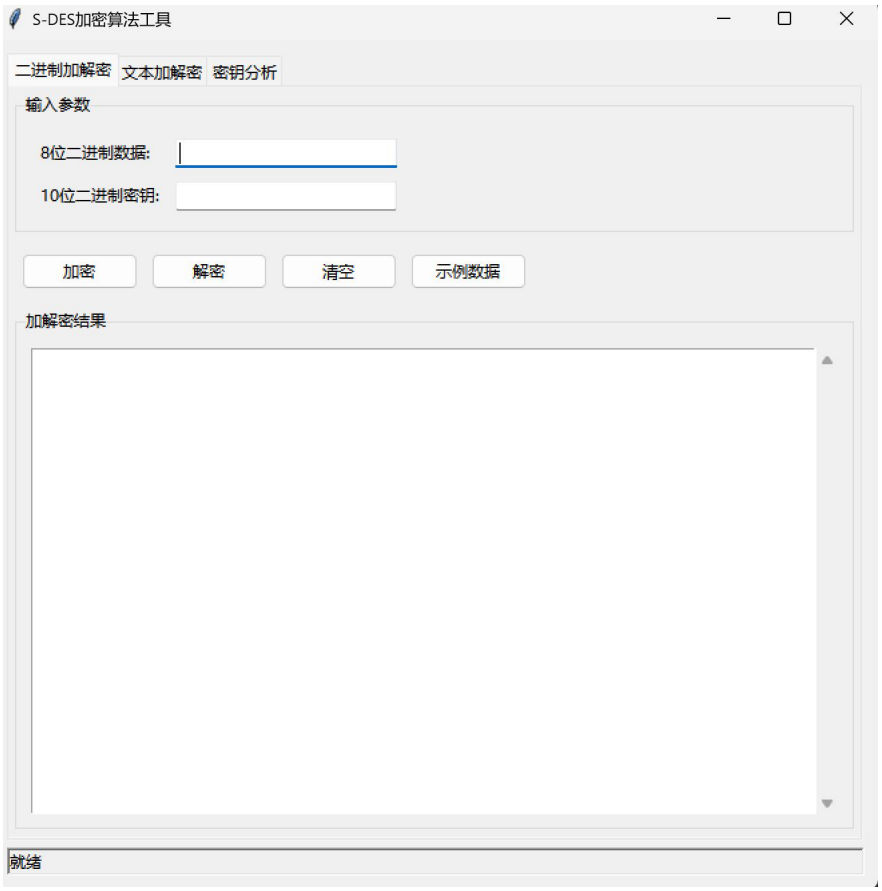


图 1 用户交互界面

四、功能模块使用方法

（一）二进制加解密

在“二进制加解密”标签页中，用户可以输入 8 位二进制明文与 10 位二进制密钥进行加密或解密。

操作步骤：

1. 输入 8 位明文（例如 10101010）；
2. 输入 10 位密钥（例如 1010000010）
3. 点击“加密”或“解密”按钮查看结果
4. 可点击“示例数据”按钮快速填入测试样例

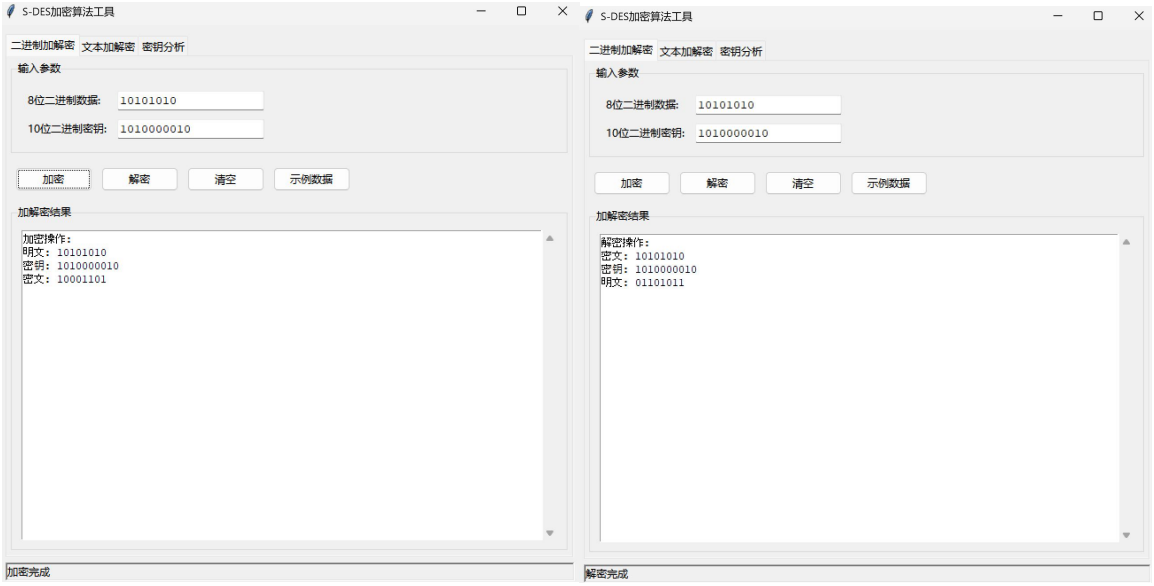


图 2 二进制加解密界面

（二）文本加解密

在“文本加解密”标签页中，用户可以输入任意文本并使用 10 位二进制密钥进行 S-DES 加密或解密。

操作步骤：

1. 输入待加密或解密的文本内容
2. 输入 10 位二进制密钥
3. 点击“加密文本”或“解密文本”按钮查看结果
4. 结果包括原文、密钥、加密结果及其十六进制表示

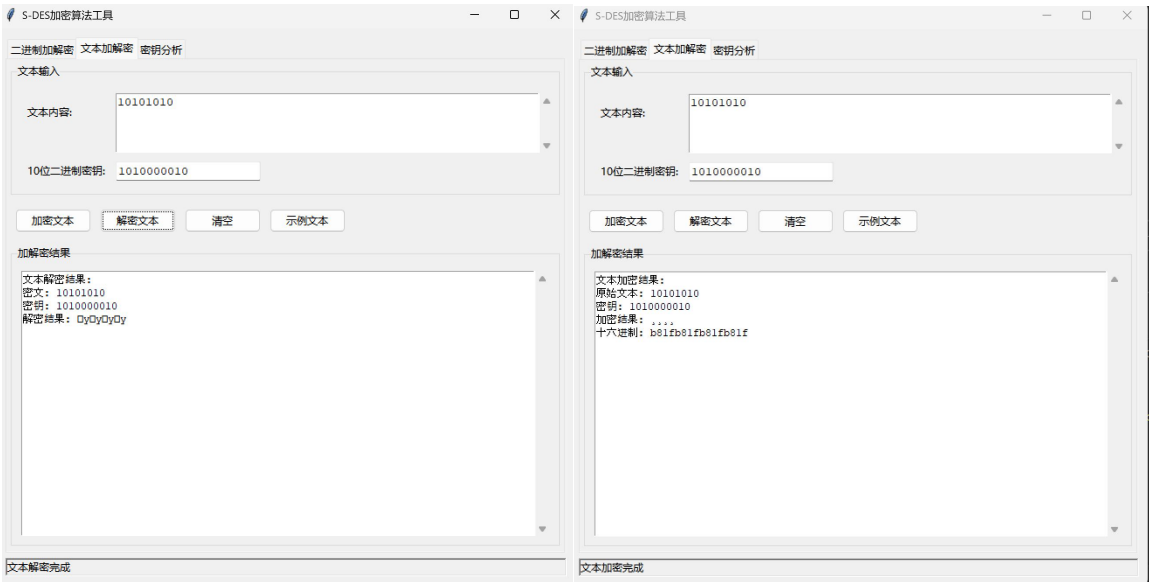


图 3 文本加解密界面

（三）密钥分析

“密钥分析”模块提供两种功能：暴力破解与密钥冲突分析。用户可通过输入已知明文和密文来模拟暴力破解，或进行多组测试分析密钥冲突。

操作步骤：

1. 在“已知明文”和“已知密文”中输入 8 位二进制数据
2. 点击“开始暴力破解”，程序遍历所有 1024 个密钥（2^{10}）进行尝试
3. 可实时查看进度条与已找到的匹配密钥
4. 可点击“停止破解”中断进程
5. “密钥冲突分析”功能会测试多个明密文对，统计产生相同密文的不同密钥数量

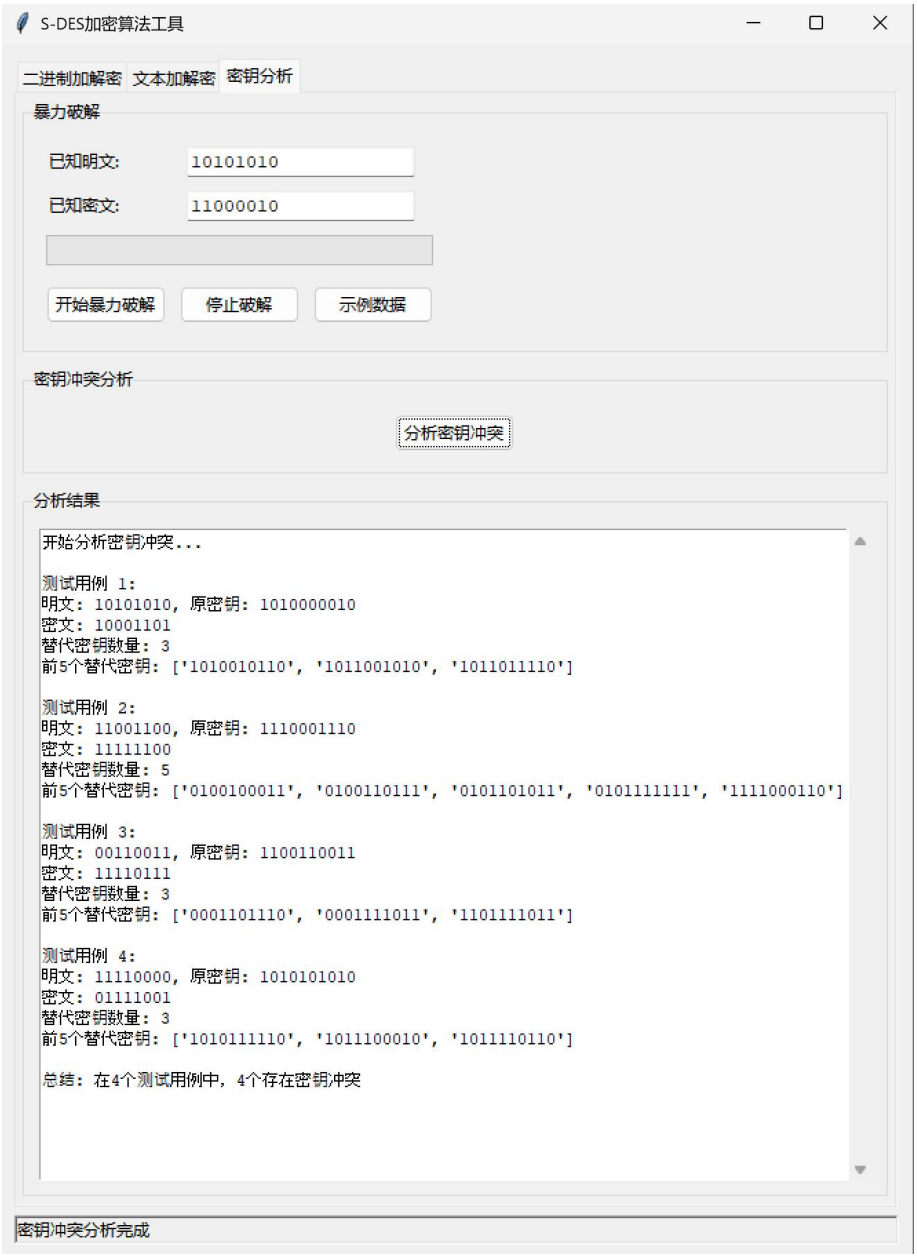
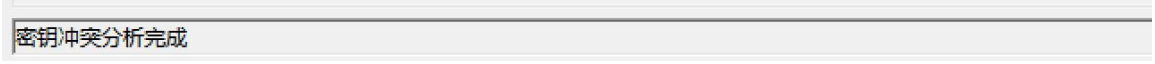


图 4 密钥分析界面

（四）状态栏

在窗口底部，软件提供状态栏，用于显示当前操作状态（如“加密完成”、“暴力破解进行中”等），方便用户了解任务执行情况。



五、常见问题与错误提示说明

1. 输入长度错误：必须输入固定长度的二进制字符串（8 位或 10 位）。

2. 非法字符：输入中只能包含‘0’或‘1’。

3. 文本加密结果乱码：S-DES 加密输出为二进制字节流，转换为字符时可能显示为不可读符号，可查看十六进制结果。

4. 暴力破解耗时较长：请耐心等待，进度条将实时更新。

六、附录：S-DES 算法原理简述

S-DES 是一种教学用途的简化版对称加密算法，通过 10 位主密钥生成两个 8 位子密钥（K1、K2），并对 8 位明文执行两轮 Feistel 结构加密。每轮包括扩展置换、S 盒替换、异或与置换操作。

核心组成包括：

- 密钥生成（P10、P8 置换与循环左移）
- 初始置换（IP）与逆置换（IP^{-1}）
- 轮函数 F（扩展置换 EP、S 盒 S0/S1、P4 置换）
- 轮间交换与异或操作