S-DES 加密解密关卡结果演示

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。

实现原理

1. **GUI 基础架构**：基于 tkinter 构建界面，包含“密钥输入框（10 位二进制限定）、明文 / 密文输入框（8 位二进制限定）、加密 / 解密按钮、结果输出区”核心组件，实现可视化交互。
2. **加解密核心逻辑**：
   - 加密函数 encrypt()：先验证输入（必须 8 位二进制），再执行 S-DES 完整流程（IP 置换→Feistel 轮函数→IP 逆置换），输出 8 位二进制密文。
   - 解密函数 decrypt()：同输入验证逻辑，逆序执行 S-DES 流程（IP 置换→逆序 Feistel 轮函数→IP 逆置换），还原 8 位二进制明文。

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。

实现原理

1. **算法参数标准化**：固定核心置换表与映射规则，所有小组统一使用，从根源避免跨工具差异：
   - 置换表：P10=[3,5,2,7,4,10,1,9,8,6]、P8=[6,3,7,4,8,5,10,9]、IP=[2,6,3,1,4,8,5,7]、IP_INV=[4,1,3,5,7,2,8,6]
   - S-Box：采用课程规定的 SBOX1、SBOX2（如 SBOX1=[[1,0,3,2],[3,2,1,0],[0,2,1,3],[3,1,3,2]]）
   - 移位规则：LEFT_SHIFT1（左移 1 位）、LEFT_SHIFT2（左移 2 位）固定定义。

第 3 关：扩展功能（ASCII 支持）

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)。

实现原理

1. **ASCII→二进制转换逻辑**:
   - 加密函数 encrypt_ascii()：遍历输入的 ASCII 字符串，通过 format(ord(char), '08b')将每个字符转为 8 位二进制（1 字节），再调用基础加密函数逐组加密，最终拼接所有密文二进制，形成完整输出。
1. **二进制→ASCII 还原逻辑**:
   - 解密函数 decrypt_ascii()：按 8 位为一组分割输入的二进制字符串，逐组调用基础解密函数得到 8 位明文二进制，通过 chr(int(decrypted_byte, 2))转为 ASCII 字符，拼接后还原原始文本。

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。

实现原理

1. **密钥穷举逻辑**:
   - 利用 2^10=1024 的有限密钥空间，通过 format(i, '010b') （i 从 0 到 1023）生成所有 10 位二进制密钥。
   - 对每个密钥，设置为当前 S-DES 密钥，加密已知明文，对比结果与已知密文，若一致则记录该密钥。
1. **进度与性能优化**:
   - 引入 progress_callback 回调，每尝试 100 个密钥更新一次进度（避免频繁刷新影响性能），同时记录开始 / 结束时间，计算总耗时。

第 5 关：封闭测试（密钥碰撞检测）

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}\neK_{j}加密得到相同密文 C_n 的情况？

实现原理

1. **随机测试数据生成**：
   - 通过 random.choice('01')生成 8 位随机明文、10 位随机原始密钥，确保测试样本的随机性与代表性。
1. **碰撞检测逻辑**：
   - 先用原始密钥加密随机明文，得到基准密文；再遍历所有其他 1023 个密钥，用相同明文加密，对比加密结果与基准密文，若一致则判定为“密钥碰撞”，记录碰撞密钥。


S-DES 算法实现
密钥 (10-bit):
1001100010
明文 (8-bit 或 ASCII):
01111001
加密
解密
暴力破解
封闭性测试
输出:
=== 封闭性测试 ===
随机明文: 01111001
原始密钥: 1001100010
原始密文: 00101100
未找到其他能产生相同密文的密钥。
☑ 结论: 在此测试中未发现密钥碰撞。
封闭性测试完成!


S-DES 算法实现
密钥 (10-bit):
1010111101
明文 (8-bit 或 ASCII):
00000001
加密
解密
暴力破解
封闭性测试
输出:
=== 封闭性测试 ===
随机明文: 00000001
原始密钥: 1010111101
原始密文: 10010110
发现 3 个不同的密钥可以产生相同密文:
0110111111
0111110111
1011010010
⚠ 结论: S-DES 不具备完美单向性，存在密钥碰撞。
封闭性测试完成!