**S-DES加密算法开发手册**

**1 系统概述**

1. DES（Simplified Data Encryption Standard）是一种简化版的对称密钥加密算法，它模拟了更复杂的DES（数据加密标准）算法的基本思想，但具有更简单的结构和较小的密钥长度。

**1.1 密钥长度**：10-bit

**1.2 分组长度：**8-bit

**1.3 加密过程**

1.3.1 密钥生成

从10位密钥生成两个子密钥（K1 和 K2），用于两个加密轮。

1.3.2 初始置换（IP）

对输入数据块进行初始置换。

1.3.3 轮函数（F）

包含非线性变换和扩展的置换，使用子密钥进行加密。

1.3.4 交换

经过两轮加密后，交换数据块的左右部分。

1.3.5 逆初始置换（IP-1）

对结果进行逆置换，得到最终的密文。

**1.4 解密过程**

解密过程与加密过程相似，但使用的子密钥顺序相反（先使用K2再使用K1）。

**1.5 安全性**

虽然S-DES比DES简单且更易于实现，但它并不适合实际应用的安全需求。S-DES的主要目的是帮助学习者理解对称加密的基本概念。

**1.6 环境要求**

IntelliJ IDEA Community Edition 2024.3.4.1

操作系统：Windows

内存：至少128MB可用内存

显示分辨率：建议1024×768或更高

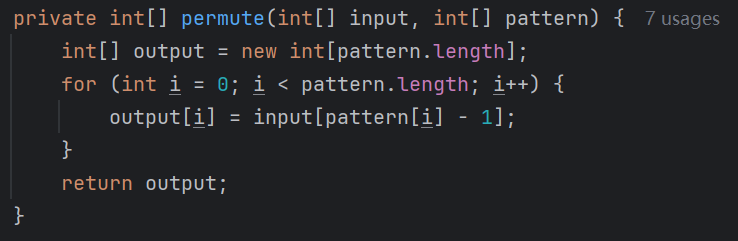
**2 代码实现**

**2.1 S-DES算法常量定义**

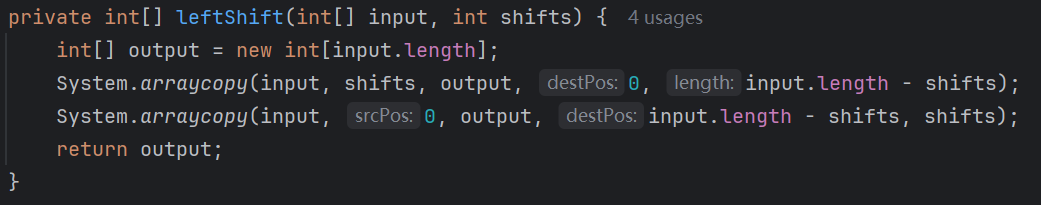


**2.2 S-DES核心算法实现**

2.2.1 置换函数



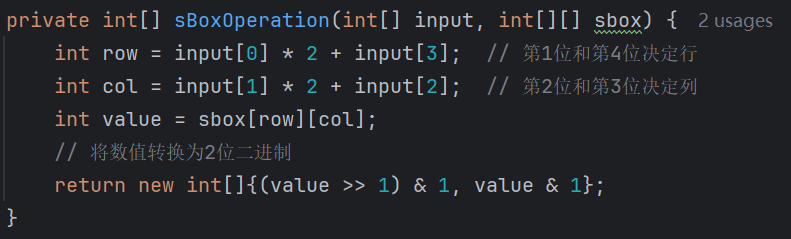
2.2.2 左循环移位函数



2.2.3 生成子密钥



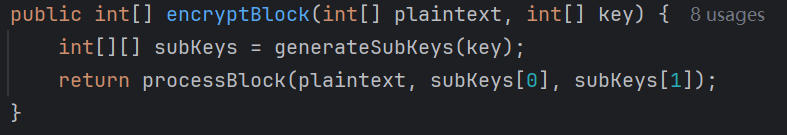
2.2.4 S盒处理函数

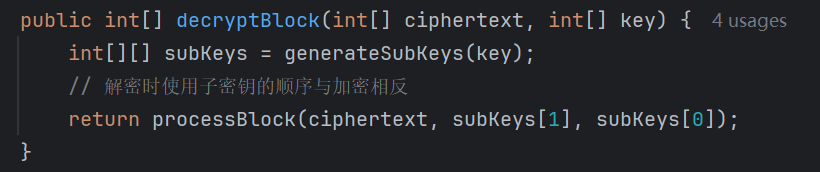


2.2.5 F轮函数

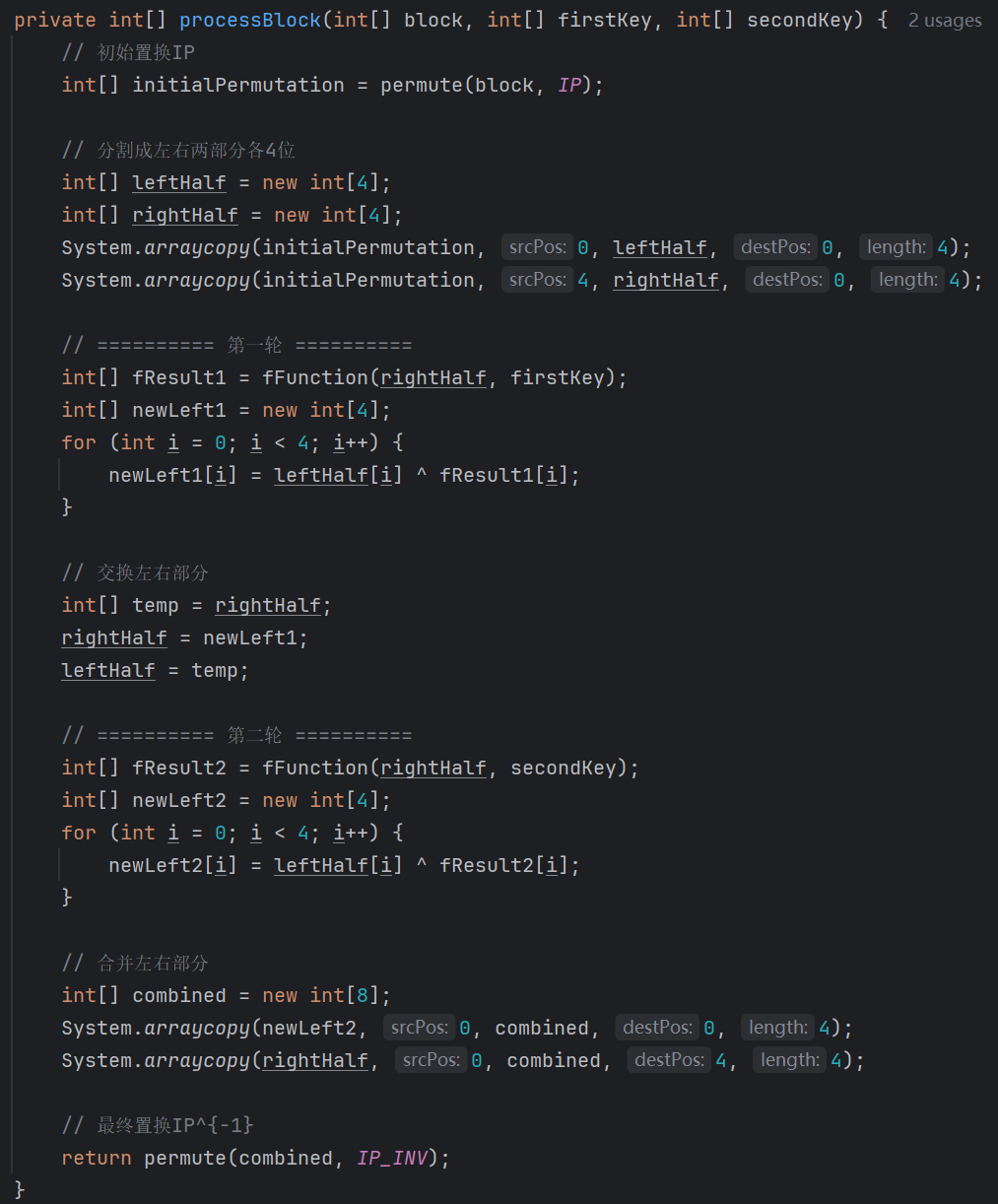


2.2.6 加密/解密单个8位分组





2.2.7 处理数据分组 (加密/解密的通用流程)



**2.3 GUI操作处理**

2.3.1 执行加密操作



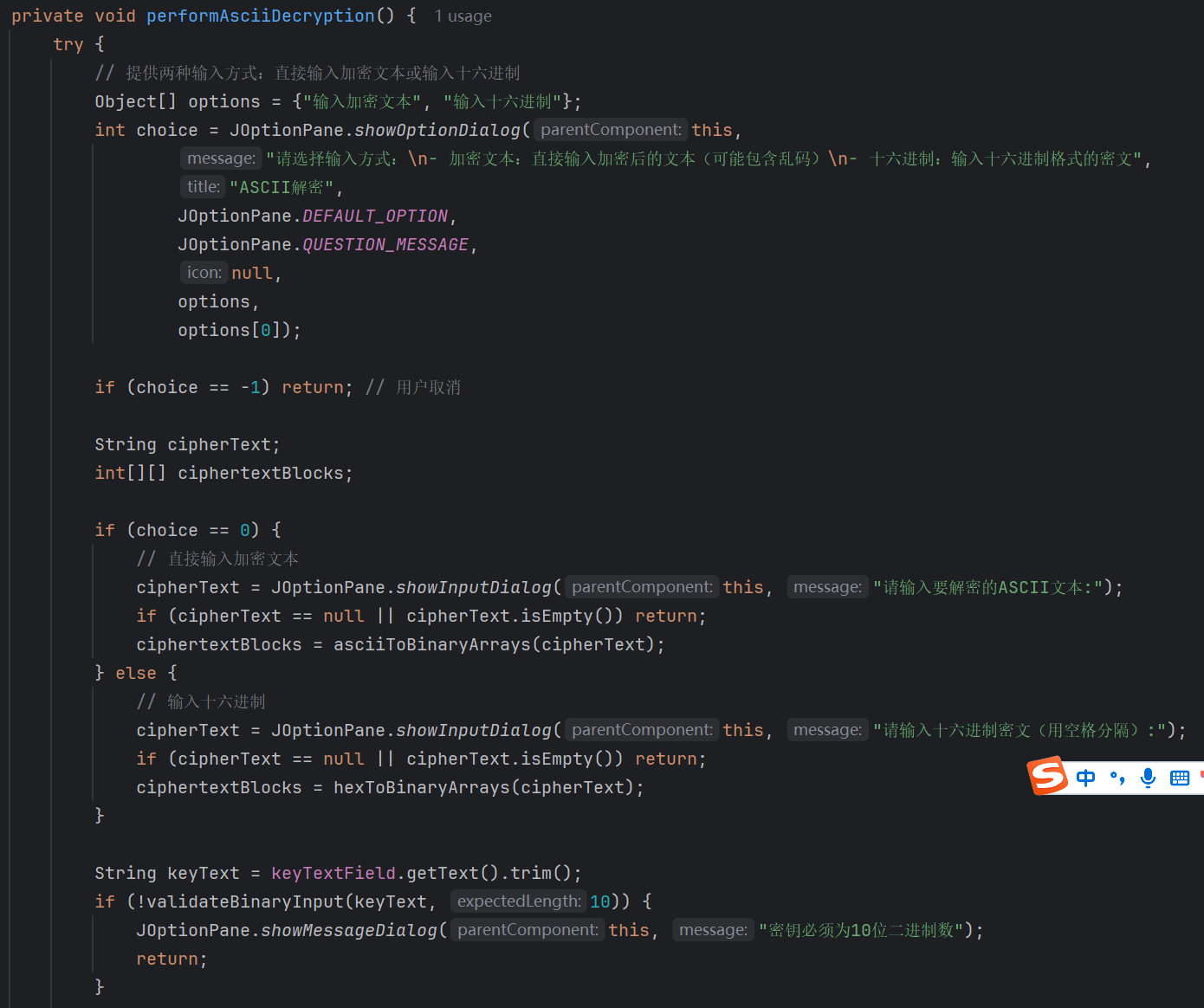
2.3.2 执行解密操作

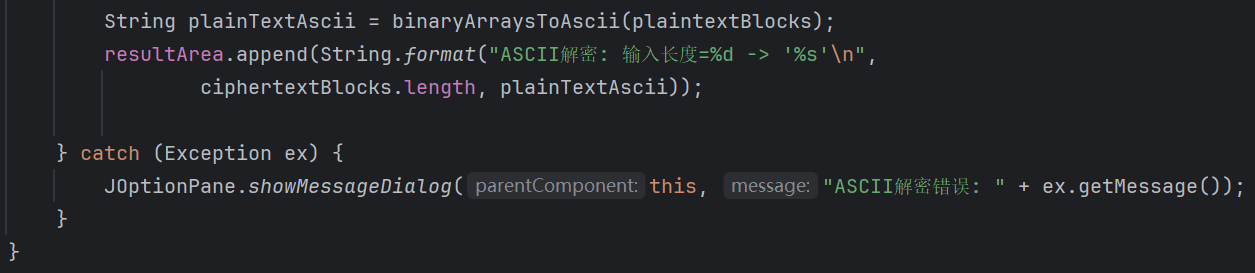


2.3.3 执行ASCII加密

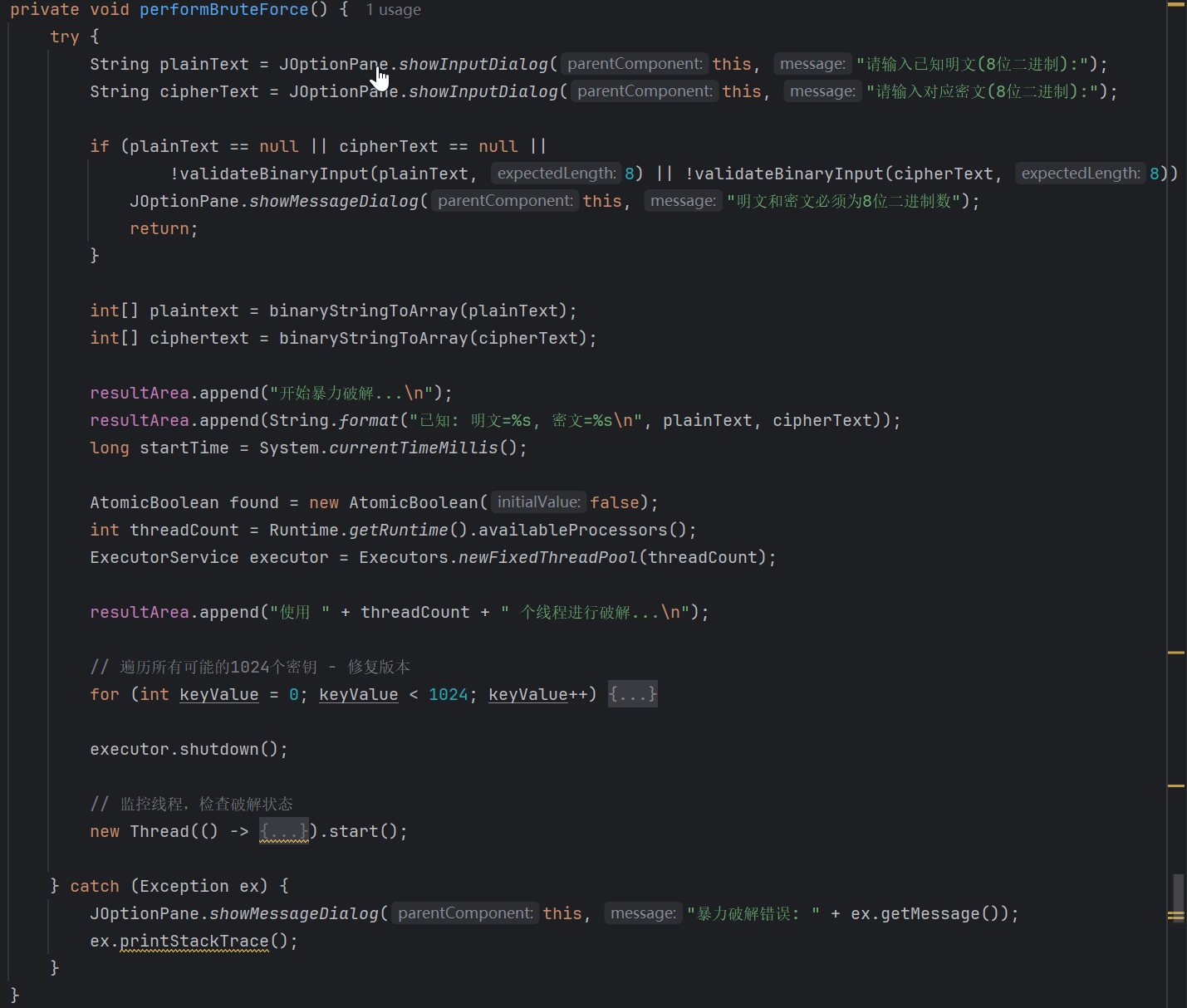


2.3.4 执行ASCII解密





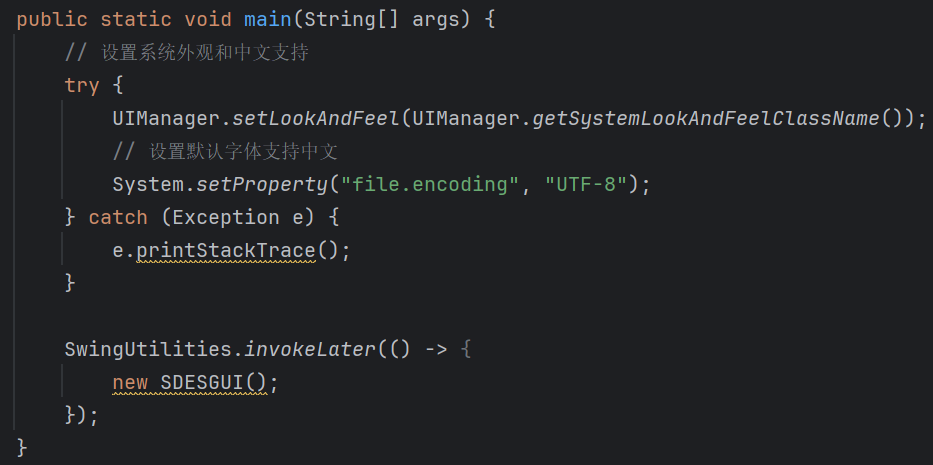
2.3.5 执行暴力破解



2.3.6 执行算法分析



**2.4 主函数**



**3 代码测试**

**3.1 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8-bit的数据和10-bit的密钥，输出是8-bit的密文。



**3.2 交叉测试**

3.2.1 加密

假设明文为10011001，密钥为1100101011，则生成的密文为11111001



3.2.2 解密

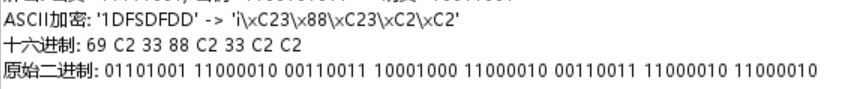
假设密文为11111001，密钥为1100101011，则生成的明文为10011001



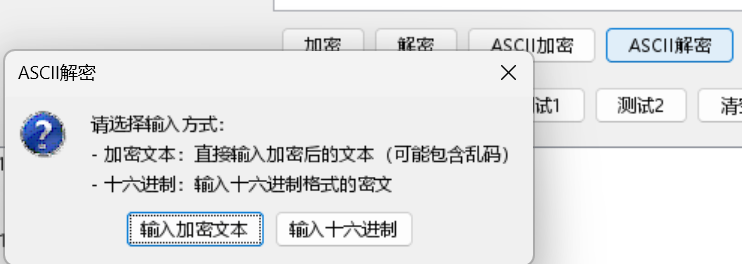
**3.3 扩展功能**

由于S-DES 处理的是 8 位二进制数据，而 ASCII 码作为表示字符的一种编码方式，它的字符通常以 7 位或 8 位二进制数表示。故运用S-DES 加密 ASCII 码时可能会出现异常。

3.3.1 ASCII加密

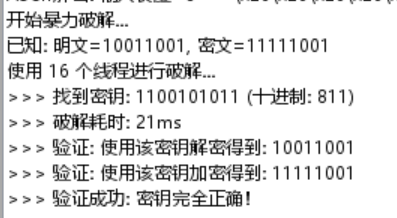


3.3.2 ASCII解密



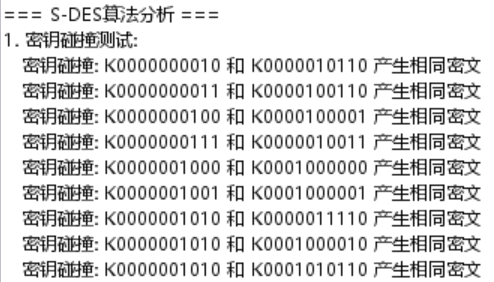


**3.4 暴力破解**



**3.5 封闭测试**

3.5.1 密钥碰撞测试



3.5.2 加解密一致性测试



**4 性能优化**

**4.1 多线程处理**

暴力破解功能使用线程池并行处理，充分利用多核CPU性能。

**4.2 内存管理**

及时释放不再使用的数组对象

合理设置线程池大小

避免内存泄漏

**5 总结**

本次实现S-DES算法的困难主要集中在几个方面：

密钥生成过程涉及复杂的置换和循环左移

S盒替换的逻辑需要精确处理行和列，且调试时难以定位问题

位操作和数据类型处理也可能导致实现上的混淆

为了提升实现的质量，采取的措施有：

将算法模块化，便于维护和理解

在输入阶段进行有效性验证，减少错误输入

编写单元测试确保各模块的正确性