DES算法的程序设计与实现

蔡梓珩 16340008

### 算法原理概述

DES是块加密。将需要加密的明文以64位一组的长度分组，通过对每一组进行初始置换、利用密钥16轮迭代、交换置换和逆置换得到64位密文。利用数学可证得，同样的对密文做相同操作可得到明文（迭代次序相反）。

### 总体结构

**加密过程：**

C = Ek(M) = IP-1·W·T16·T15·…·T1·IP(M)

* M为算法输入的64位明文块；
* Ek描述以K为密钥的加密函数，由连续的过程复合构成；
* IP为64位初始置换；
* T1，T2，…，T16是一系列的迭代变换；
* W为64位置换，将输入的高32位和低32位交换后输出；
* IP-1是IP 的逆置换；
* 是IP 的逆置换。

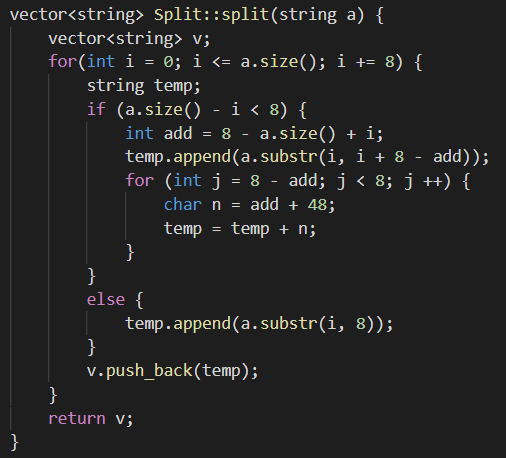
**解密过程：**

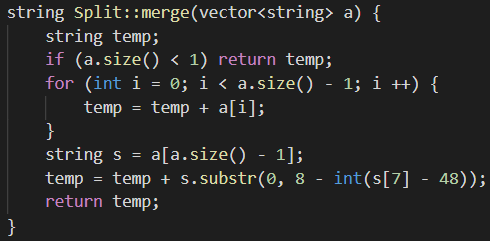
M = Dk(C) = IP-1·W·T1·T2·…·T16·IP(C)

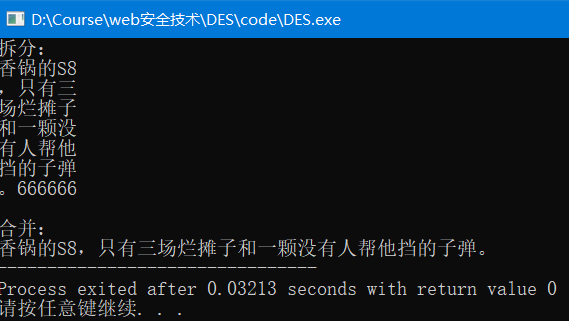
### 模块分解：

1. 信息空间

首先需要将明文分割成等长64位的字符串。最后一组若不足8字节则补全（用要补全的个数取值），若刚好为64位，则再添加一个一个分组，里面8个字节都取8。这里编了一个Split类用于分割以及合并明文。

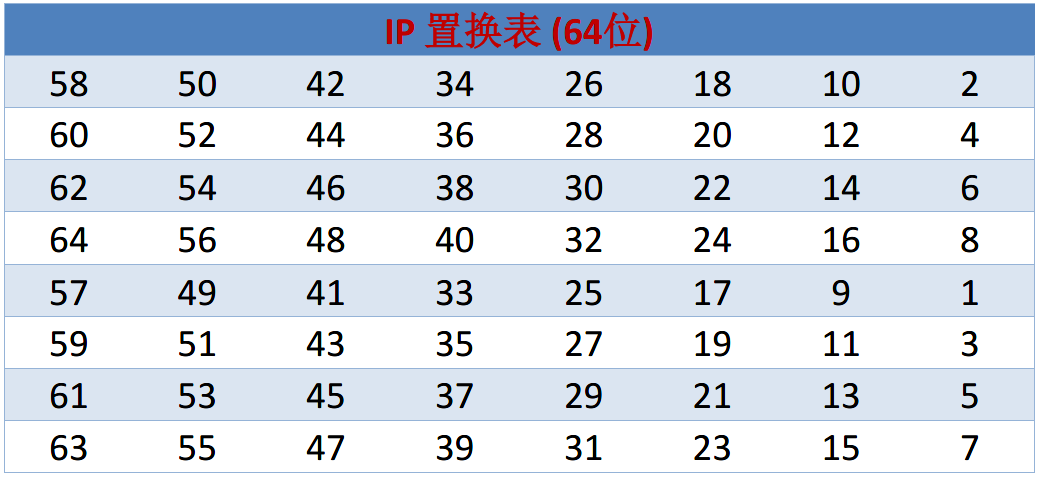
分割实现：

合并实现：

测试效果：

### IP置换

给定给定64位明文块M，通过一个固定的初始置换IP 来重排M 中的二进制位，得到二进制串M0 = IP(M) = L0R0，这里L0 和R0分别是M0 的前32位和后32位。下表给出IP 置换后的下标编号序列。



这里的IP置换表的第a个数字，相当于新数组第a个位，其中数字i相对于旧数组的num[i]。即表中数字相当于这个位置的数字在旧数组中的下标。因此将此表录入数组，然后即可根据下标置换。这里为了方便转换操作，将64位的8字节字符串换成64字节对应其二进制值。