1 ZUC S0 和 S1 的差分分析表和线性分析表的实现

源代码：见 test.py

设计思路:

DDT:将 S 盒的输入输出转换为八位二进制，将输入的八位二进制字符串分为前后两部分以便于在 S 盒中查找输出结果。对于任意输入 x 及差分Δx，有 x′=x+Δx，根据 x 和 x′有对应输出 y 和 y′并可获取对应Δy，对于 SPN，Δx 和Δy 可取 0 到 255,将差分结果填入 256×256 的表中则为 DDT。

LAT：对于输入 X=x1x2x3x4x5x6x7x8,Y=y1y2y3y4y5y6y7y8,及输入输出系数(x,y)计算不同输入输出使输入输出系数对应的异或表达式等于 0 的次数 N（x,y）,将 N(x,y)-128 作为结果填入输入输出系数 256×256 的表格中。(举例，假设输入输出系数为（3，4）,则计算使得 x7⊕x8⊕y6=0 的输入输出次数，再减去 128 则为 lat 中的结果)

实验结果：见 latofzuc_s0, latofzuc_s1,ddtofzuc_s0, ddtofzuc_s1

2 为什么加密者需要最后一轮密钥混合？以 SPN 为例

答:线性分析可以通过构造一个关于明文和倒数第二轮输入的线性表达式恢复出最后一轮加密使用的子密钥的一个子集，因此需要最后一轮密钥混合。