++++++++++++++++++++++++++++++++++++++

测试用例一：

1）输入参数：

服务商机构代码< C1234567 > 订单编号< 20250515134567abcdefghijk > 金额< 5000 >

服务商 key：9BEF0B09ADF351AF81DCBCFD276C3CAF

2）输出结果（pidSct 对应 ASCII 码）：

8B607024

++

详细计算步骤（a-h）：

a) 将（服务商 PID+订单编号+订单金额）形成 MAC ELEMEMENT BLOCK（MAB）

MAB: C1234567 20250515134567abcdefghijk 000000005000 长度: 63

b) SM4 算法的 MAB，按每 16 个字节做异或，如果最后不满 16 个字节，则添加 0x00。

MAB 分组：[b'C1234567 20250', b'515134567abcdefg', b'hijk ', b'000000005000\x00']

XOR 运算开始：

输入块 1: 01000011 00110001 00110010 00110011 00110100 00110101 00110110 00110111 00100000 00100000 00100000 00110010 00110000 00110010 00110101 00110000

输入块 2: 00110101 00110001 00110101 00110001 00110011 00110100 00110101 00110110 00110111 01100001 01100010 01100011 01100100 01100101 01100110 01100111

输出块 : 01110110 00000000 00000111 00000010 00000111 00000001 00000011 00000001 00010111 01000001 01000010 01010001 01010100 01010111 01010011 01010111

XOR 运算开始：

输入块 1: 01110110 00000000 00000111 00000010 00000111 00000001 00000011 00000001 00010111 01000001 01000010 01010001 01010100 01010111 01010011 01010111

输入块 2: 01101000 01101001 01101010 01101011 00100000 00100000 00100000 00100000 00100000 00100000 00100000 00100000 00100000 00100000 00100000 00100000

输出块 : 00011110 01101001 01101101 01101001 00100111 00100001 00100011 00100001 00110111 01100001 01100010 01110001 01110100 01110111 01110011 01110111

XOR 运算开始：

输入块 1: 00011110 01101001 01101101 01101001 00100111 00100001 00100011 00100001 00110111 01100001 01100010 01110001 01110100 01110111 01110011 01110111

输入块 2: 00100000 00100000 00100000 00110000 00110000 00110000 00110000 00110000 00110000 00110000 00110101 00110000 00110000 00110000

00000000
输出块 ： 00111110 01001001 01001101 01011001 00010111 00010001 00010011 00010001 00000111 01010001 01010010 01000100 01000100 01000111 01000011 01110111
---
c) 将异或运算后的最后 16 个字节（RESULT BLOCK）转换成 32 个 HEXDECIMAL
异或结果 RESULT BLOCK HEXDECIMAL: 3E494D59171113110751524444474377
结果分组（每组 16 字节，按 ASCII 字符显示）：
组 1: 3E494D5917111311 --用来做 SM4 加密
组 2: 0751524444474377 --用来与组 1 数据 SM4 加密结果做异或

d)、
组 1 SM4 加密：
data： 33453439344435393137313131333131
key：9BEF0B09ADF351AF81DCBCFD276C3CAF
result1：
7C9F6E83960C549AE503F699A605D2E1

e)、
result1 xor 组 2：
7C9F6E83960C549AE503F699A605D2E1 xor 30373531353234343434343734333737
result2:
4CA85BB2A33E60AED137C2AE9236E5D6

f)、
再次 sm4 加密：
result2： 4CA85BB2A33E60AED137C2AE9236E5D6
key：9BEF0B09ADF351AF81DCBCFD276C3CAF
result3:
8B607024746D5866188CBBFD284BEF14

g)、
to asicc:
8B607024746D5866188CBBFD284BEF14->
%H38%H42%H36%H30%H37%H30%H32%H34%H37%H34%H36%H44%H35%H38%H36%H36
%H31%H38%H38%H43%H42%H42%H46%H44%H32%H38%H34%H42%H45%H46%H31%H34

h)、取结果前 8 位作为 tusn mac 结果：
%H38%H42%H36%H30%H37%H30%H32%H34

++++++++++++++++++++++++++++++++++++++

测试用例二：

1）输入参数：

服务商机构代码< C1009999 > 订单编号< 12202503210345024838751O4 > 金额< 123 >

服务商 key：F1774E4EDB4396B1647304EB4A6A4E3D

2）输出结果（**pidSct 对应 ASCII 码)**：

D36A3239

++++++++++++++++++++++++++++++++++++

详细计算步骤（a-h)：

a) 将（服务商 PID+订单编号+订单金额）形成 MAC ELEMEMENT BLOCK（MAB）

MAB: C1009999 12202503210345024838751O4 000000000123 长度: 63

b) SM4 算法的 MAB，按每 16 个字节做异或，如果最后不满 16 个字节，则添加 0x00。

MAB 分组: [b'C1009999 12202', b'5032103450248387', b'5104 ', b'000000000123\x00']

XOR 运算开始：

输入块 1: 01000011 00110001 00110000 00110000 00111001 00111001 00111001 00111001 00100000 00100000 00100000 00110001 00110010 00110010 00110000 00110010

输入块 2: 00110101 00110000 00110011 00110010 00110001 00110000 00110011 00110100 00110101 00110000 00110010 00110100 00111000 00110011 00111000 00110111

输出块 : 01110110 00000001 00000011 00000010 00001000 00001001 00001010 00001101 00010101 00010000 00010010 00000101 00001010 00000001 00001000 00000101

XOR 运算开始：

输入块 1: 01110110 00000001 00000011 00000010 00001000 00001001 00001010 00001101 00010101 00010000 00010010 00000101 00001010 00000001 00001000 00000101

输入块 2: 00110101 00110001 00110000 00110100 00100000 00100000 00100000 00100000 00100000 00100000 00100000 00100000 00100000 00100000 00100000 00100000

输出块 : 01000011 00110000 00110011 00110110 00101000 00101001 00101010 00101101 00110101 00110000 00110010 00100101 00101010 00100001 00101000 00100101

XOR 运算开始：

输入块 1: 01000011 00110000 00110011 00110110 00101000 00101001 00101010 00101101 00110101 00110000 00110010 00100101 00101010 00100001 00101000 00100101

输入块 2: 00100000 00100000 00100000 00110000 00110000 00110000 00110000 00110000 00110000 00110000 00110000 00110000 00110001 00110010 00110011 00000000

输出块 ： 01100011 00010000 00010011 00000110 00011000 00011001 00011010 00011101 00000101 00000000 00000010 00010101 00011011 00010011 00011011 00100101

---

c) 将异或运算后的最后 16 个字节（RESULT BLOCK）转换成 32 个 HEXDECIMAL
异或结果 RESULT BLOCK HEXDECIMAL: 6310130618191A1D050002151B131B25
结果分组（每组 16 字节，按 ASCII 字符显示）：
组 1: 6310130618191A1D --用来做 SM4 加密
组 2: 050002151B131B25 --用来与组 1 数据 SM4 加密结果做异或

d)
组 1 SM4 加密：
data： 36333130313330363138313931413144
key：F1774E4EDB4396B1647304EB4A6A4E3D
result1：
C1B3C487A397302770C7FD74FCD3F010

e)
result1 xor 组 2：
C1B3C487A397302770C7FD74FCD3F010 xor 30353030303231353142313331423235
result2:
F186F4B793A501124185CC47CD91C225

f)
再次 sm4 加密：
result2： F186F4B793A501124185CC47CD91C225
key：F1774E4EDB4396B1647304EB4A6A4E3D
result3:
D36A3239A2BB44AF6ED0A56C11A708BF

g)
to asicc:
D36A3239A2BB44AF6ED0A56C11A708BF->
44 33 36 41 33 32 33 39 41 32 42 42 34 34 41 46
36 45 44 30 41 35 36 43 31 31 41 37 30 38 4246

h) pidSct 的计算结果：
取结果前 8 位作为 tusn mac 结果：
%H44%H33%H36%H41%H33%H32%H33%H39（D36A3239）