实验报告（SM4 及工作模式）

【实验目的】

1、 通过本次实验，熟练掌握 SM4 加解密流程；
2、 通过本次实验，了解并掌握各种工作模式；
3、 感受工作模式与填充方式对安全性的意义。

【实验环境】

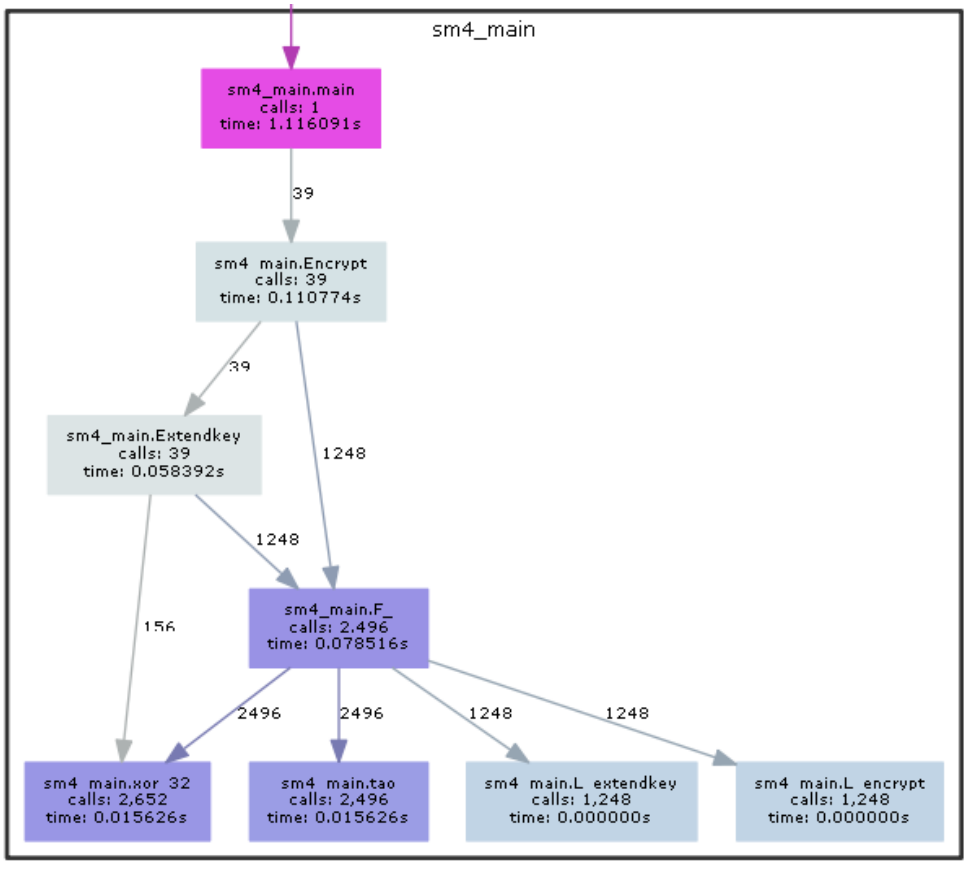
- Python 3.9.7 64-bit

【实验内容】

一、SM4 算法

1、算法流程

函数调用关系图如下（以 39 次加密为例）：



函数名	函数功能
Extendkey()	密钥拓展
xor_32()	32 bit 异或
F_()	轮函数F
tao()	非线性变换τ
L_extendkey()	密钥拓展的线性变换L
L_encrypt	加/解密的线性变换L

2、测试样例及结果截图

```
39
0x9241921686391cf123780b56ced56fdc
0xe97c4d46e8554da3a8eb618891ec5211
0
0xf2a8ae2ce42146c1491f872daafb110e

Process finished with exit code 0
```

```
1
0x1f47f42618704f95e61da82daf959bd8
0x3b3678c215121f0e837e04f58584ed32
1
0xf3bf692c7a33557035a115624ef5f976

Process finished with exit code 0
```

```
1
0xc04a9b311a2fc245f742c5719fcf249d
0x4046fb1985d94a7f1ff55ec7ec5f6054
0
0x6b956ddb0faff373bc338cb600739f23

Process finished with exit code 0
```

```
39
0x52a4fd41fee4f0bacfb0a3347b52f55d
0x0b93875661edea7fcd44c8643c76cfc6
1
0x8dd5a004b00ff3c9eb6bc513964e7d3b

Process finished with exit code 0
```

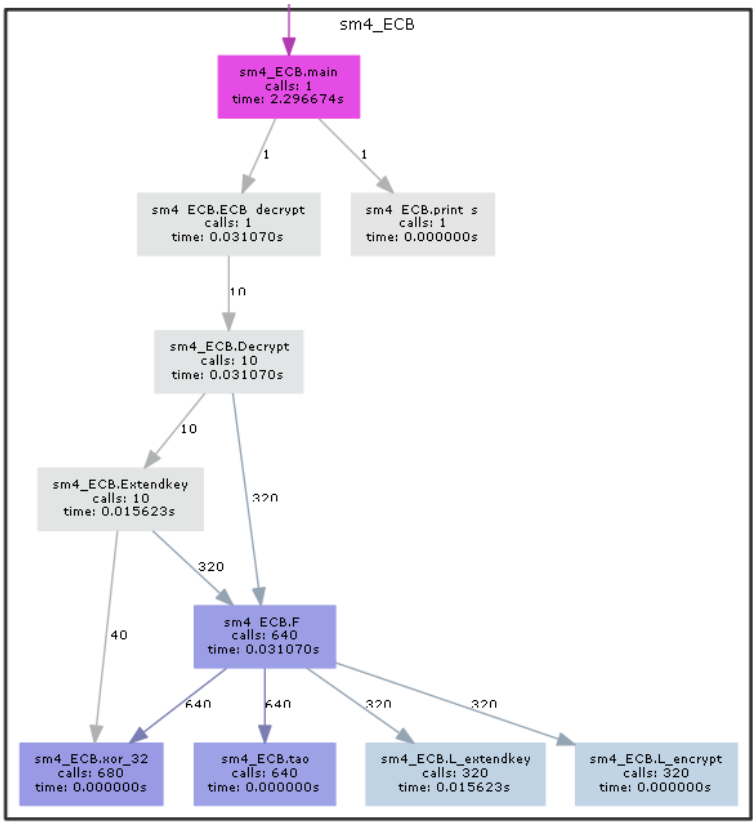
3、讨论与思考

- 该程序在上学期的数电实验中写过，因此相对比较轻松，里面的格式处理基本用的都是'{}'.format()。

二、SM4 - ECB 模式

1、算法流程

函数调用关系图如下（以解密为例）：



函数名	函数功能
ECB_decrypt()	ECB 模式下的解密
print_s()	加/解密结果输出

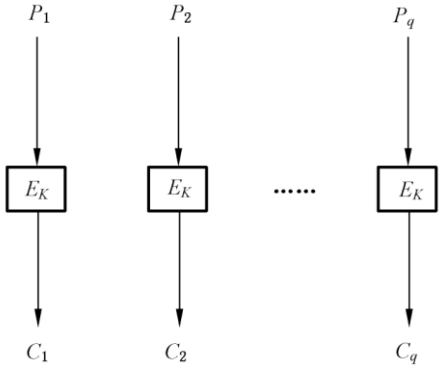
2、测试样例及结果截图（结果过多，仅输出一组及 OJ 截图）

```
0x4c 0xe0 0xa1 0x25 0xd5 0x64 0x2e 0xb5 0xa5 0x6b 0xcb 0x4e 0x7b 0x85 0x0a 0x49
0x43 0xe8 0xa2 0x23 0xd4 0x19 0xe5 0x4d 0x5b 0xe6 0xa5 0x19 0x98 0xa6 0x84 0x67
0xd3 0x4b 0xa1 0xb0 0xa7 0x89 0xa4 0x69 0x1c 0xf8 0x99 0xde 0x2e 0x7c 0x2d 0x75
0xa6 0xf6 0x61 0x99 0xe2 0xb1 0xca 0x50 0x15 0x49 0x88 0x1f 0x92 0xc4 0x35 0xb6
0xae 0x29 0xf0 0x71 0x13 0xed 0x9f 0x08 0x13 0xf2 0x3f 0x64 0x5f 0xa1 0x38 0xbe
0x24 0xb8 0x06 0xaa 0x20 0x6e 0x89 0xd0 0x55 0xf1 0x30 0x84 0xb9 0xfb 0xd9 0x65
0x17 0x06 0xac 0x40 0xd8 0xf4 0x03 0x49 0x26 0xdc 0xdd 0x2b 0x0f 0xd2 0x19 0x2d
0xcf 0x2d 0xe7 0x87 0xcf 0xd6 0x14 0x49 0x89 0xf9 0x79 0x4f 0xb9 0xbb 0x8d 0xd9
0x1c 0x50 0x0c 0x33 0xc2 0xc5 0x5b 0xdb 0xd3 0xc5 0x1c 0x77 0xbb 0x19 0x12 0x8f
0xb6 0x60 0x24
Process finished with exit code 0
```

评测编号	提交时间	提交状态	代码语言	最大运行时间	最大运行内存	详细信息
15530	2022-04-21 10:41:17	Accepted	Python	98ms	8732KB	

3、讨论与思考

- ECB——电码本（Electronic CodeBook）
- ECB 模式将整个明文分成若干段相同的小段，然后对每一小段进行加密：



三、SM4 - CBC 模式

1、算法流程

伪代码：

Algorithm 1：SM4 — CBC Encrypt

Input: plaintext P, initial vector IV, key K

Output: ciphertext C

1. **function** "SM4_CBC_Enc(P, IV, K)"
2. /* *grouping & padding* */
3. $[P_1, P_2, \cdots, P_q] \leftarrow P$
4. $C_1 \leftarrow E_K(P_1 \oplus IV)$

5. **set** $i \leftarrow 2$
6. **while** $i \leqslant q$ **do**
7. $C_i \leftarrow E_K(P_1 \oplus IV)$
8. $i \leftarrow i+1$
9. **end while**
10. $C \leftarrow [C_1, C_2, \cdots C_q]$
11. **return** C
12. **end function**

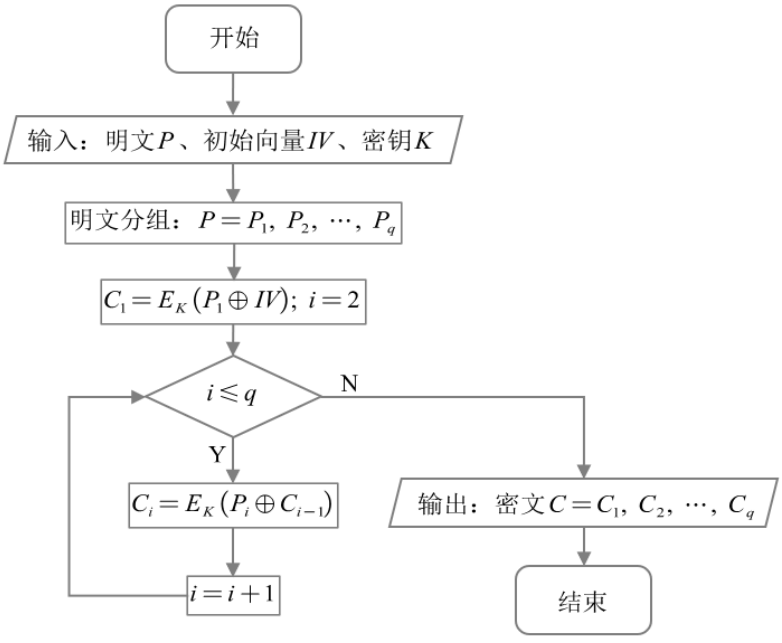
Algorithm 2：SM4 − CBC Decrypt

Input: ciphertext C, initial vector IV, key K

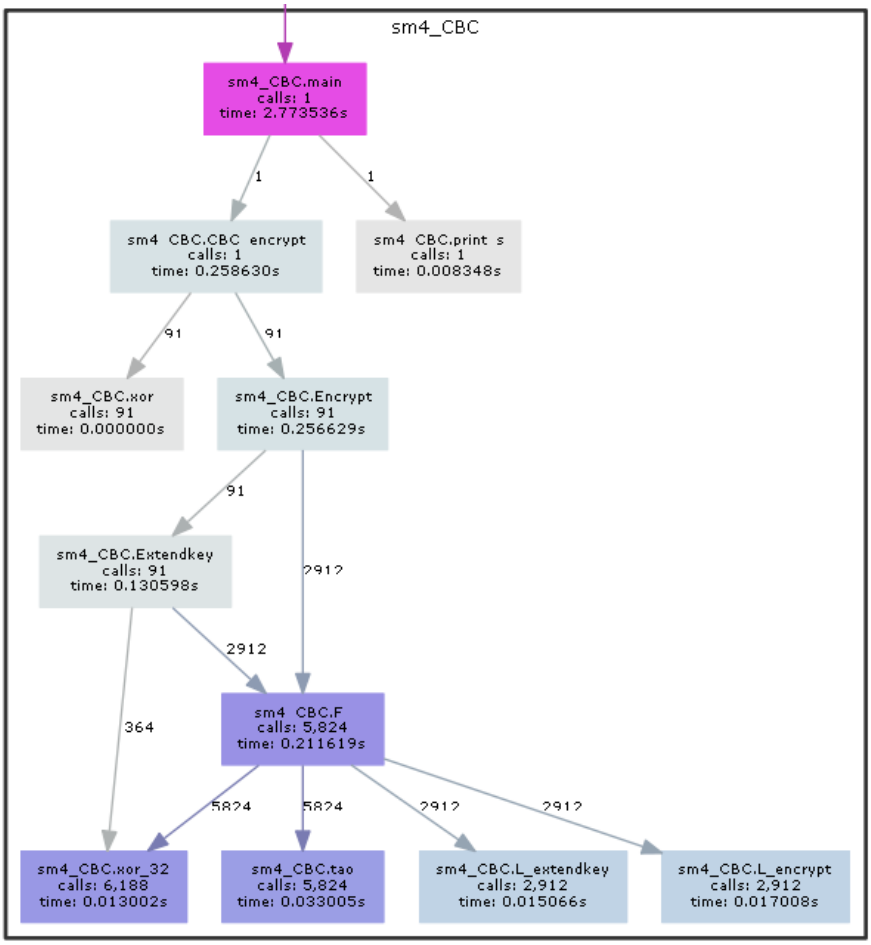
Output: plaintext P

1. **function** "SM4_CBC_Dec$(C,\ IV,\ K)$"
2. / * *grouping & padding* * /
3. $[C_1, C_2, \cdots, C_q] \leftarrow C$
4. $P_1 \leftarrow D_K(C_1) \oplus IV$
5. **set** $i \leftarrow 2$
6. **while** $i \leqslant q$ **do**
7. $P_i \leftarrow D_K(C_i) \oplus C_{i-1}$
8. $i \leftarrow i+1$
9. **end while**
10. $P \leftarrow [P_1, P_2, \cdots P_q]$
11. **return** P
12. **end function**

流程图：



函数调用关系图如下（以解密为例）：



2、测试样例及结果截图（结果过多，仅输出一组及 OJ 截图）

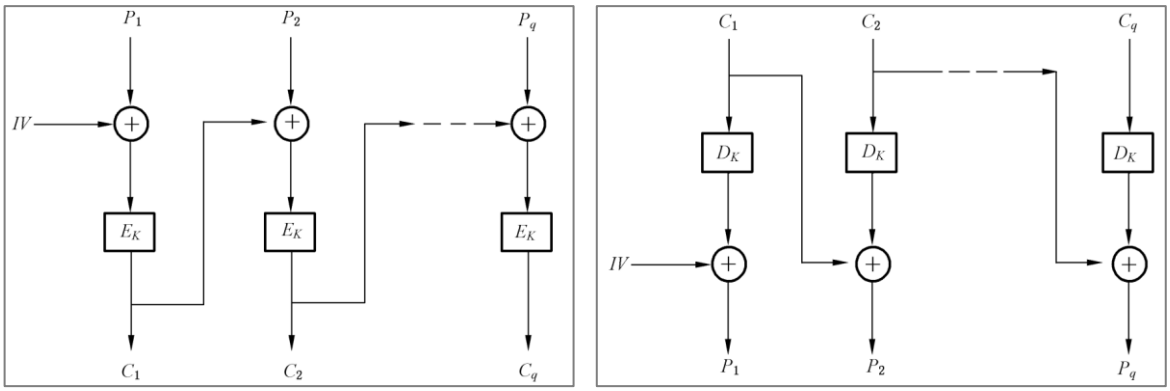
```
0xff 0xa3 0x0c 0xe7 0x5f 0xb7 0x68 0x93 0xd4 0xec 0x43 0x6a 0xfe 0xbb 0xd4 0x24
0xc8 0xd5 0xc7 0x56 0x88 0x71 0x2c 0x17 0x77 0xae 0x14 0x40 0x52 0xec 0x08 0x73
0xb9 0xdb 0x64 0xf8 0x5a 0xc4 0x45 0xe5 0x4e 0x50 0x8f 0x0c 0xf0 0x07 0xdd 0x5d
0xd2 0xce 0xfd 0x5a 0xb0 0x3d 0x05 0xe8 0x70 0x78 0xdc 0x49 0xec 0x59 0x91 0x5a
0xd2 0xe6 0xe0 0x84 0xe1 0x21 0x76 0x7f 0x21 0x8a 0x30 0xad 0xfb 0x11 0x0d 0x27
0xf1 0x86 0x17 0x1a 0x14 0x83 0x2a 0x30 0x77 0x5a 0x3f 0xb8 0x0a 0x73 0x6c 0xed
0x79 0xdc 0xc6 0xa8 0x1d 0xb2 0xfb 0xef 0x5a 0xbf 0x4f 0x36 0x05 0x32 0xd7 0xae
0x02 0xcf 0xaf 0xc6 0x45 0xac 0xb2 0xc9 0x43 0x7d 0x12 0x45 0x5a 0xe8 0xba 0xbf
0x53 0xef 0x66 0xa9 0x76 0x6c 0x6a 0xf7 0xa5 0xf4 0x56 0xe3 0x6d 0x4a 0x83 0x0b
0x64 0x09 0xcc 0x89 0x71 0x8c 0xf0 0x10 0xed 0x65 0x66 0xe5 0x79 0xa4 0x9e 0xed

Process finished with exit code 0
```

评测编号	提交时间	提交状态	代码语言	最大运行时间	最大运行内存	详细信息
15799	2022-04-21 12:10:05	Accepted	Python	107ms	8728KB	

3、讨论与思考

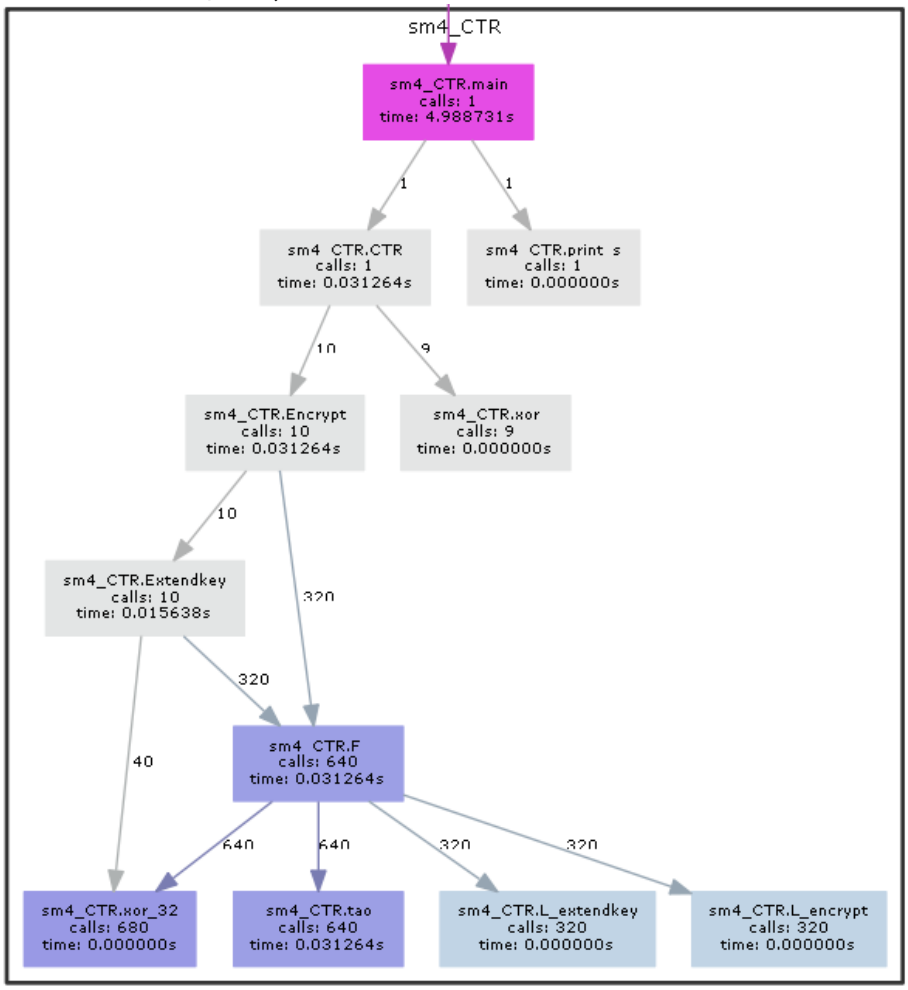
- CBC——密文分组链接（Cipher Block Chaining）
- CBC 模式先将明文切分成若干小段，然后每一小段与初始块或者上一段的密文段进行异或运算后，再与密钥进行加密；解密先与密钥解密，结果与初始块或上一段迷密文异或得到对应明文块：



四、SM4 - CTR 模式

1、算法流程

函数调用关系图如下（以加密为例，但实际上都一样）：



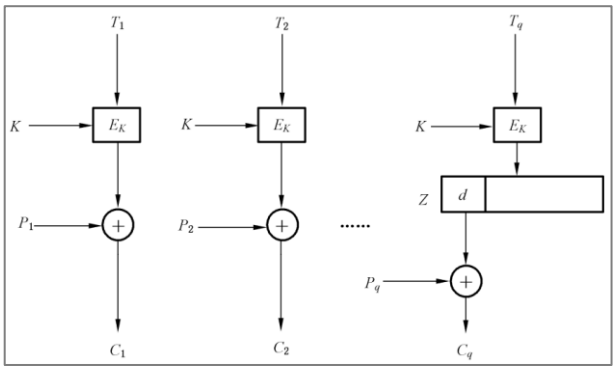
2、测试样例及结果截图（结果过多，仅展示 OJ 截图及一组输出）

评测编号↓	提交时间	提交状态	代码语言	最大运行时间	最大运行内存	详细信息
16306	2022-04-22 00:24:37	Accepted	Python	99ms	8732KB	

```
0xe1 0x69 0x81 0xb7 0x38 0xd9 0x28 0x75 0xe0 0xaf 0x6b 0xc8 0x17 0xec 0x39 0xae
0xb4 0x22 0x68 0xe0 0xd2 0x5b 0x94 0xd7 0xa8 0xe0 0xd5 0xe7 0xcd 0x86 0x9c 0x54
0x5e 0x9d 0xc3 0x6e 0x73 0x06 0x5b 0x13 0xd5 0x18 0x51 0xe6 0xa3 0xa6 0x94 0xaa
0xbf 0x95 0xeb 0x06 0x0b 0x12 0xeb 0xa8 0x89 0xc2 0x71 0xac 0xb4 0xdd 0x67 0xc6
0xea 0x5a 0xd5 0x8a 0x58 0x06 0x2b 0xc1 0x5a 0x42 0x25 0x7b 0x9a 0xc6 0x5d 0x7b
0xd9 0xd8 0xc2 0x39 0xa5 0xcc 0xbe 0x51 0xac 0x10 0x4e 0x1d 0x6f 0xf6 0x33 0x01
0x79 0x0a 0x02 0x42 0xd8 0x05 0xdd 0x7d 0x8f 0xb5 0x5a 0x81 0xd9 0x69 0x98 0xd5
0x65 0xf5 0x63 0x2f 0x1e 0xfe 0xca 0xff 0x02 0x93 0x38 0x0a 0x9e 0xea 0x0d 0x0c
0xc3 0x9e 0x4b 0x1a 0xac 0xf2 0x73 0x3f 0x83 0xbd 0xc3 0x4c 0x7b 0x75 0xc7 0xcb
0x0b 0x4e
Process finished with exit code 0
```

3、讨论与思考

- CTR——计数器（Counter）
- CTR 模式不常见，在 CTR 模式中，有一个自增的算子，这个算子用密钥加密之后的输出和明文异或的结果得到密文，相当于一次一密。这种加密方式简单快速，安全可靠，而且可以并行加密，但是在计算器不能维持很长的情况下，密钥只能使用一次。CTR 的示意图如下所示：

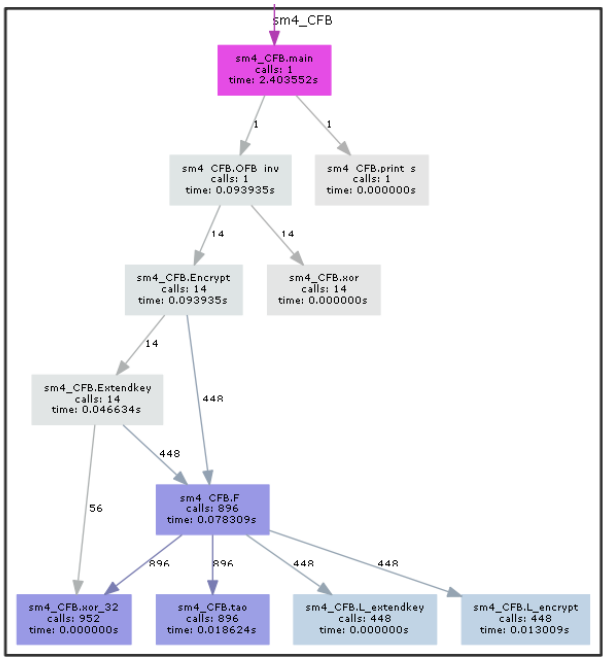


- CTR 模式没有 padding，直接取最后一次加密后的前对应位数异或明文即可。
- CTR 模式加/解密没有本质区别。

五、SM4 - CFB 模式

1、算法流程

函数调用关系图如下（以解密为例）：



2、测试样例及结果截图（结果过多，仅展示一组输出及 OJ 截图）

```
0x36 0x3a 0x3d 0x07 0x65 0xdc 0xf2 0x04 0xc7 0x96 0x7c 0x17 0x5a 0x04 0xb5 0x8d
0x01 0x4d 0x21 0x47 0x99 0x89 0xfe 0x7a 0x57 0x44 0x04 0x81 0xb8 0x91 0x2d 0xc9
0x87 0xb3 0x2e 0x71 0x43 0x6c 0x7f 0x0d 0x34 0xfc 0xaa 0x61 0x66 0x35 0x66 0xe0
0x26 0x80 0xa9 0x92 0xe8 0xdc 0xe2 0xb7 0x5a 0x36 0x9c 0xbc 0x72 0x51 0x9e 0x73
0x99 0x72 0xb2 0xaf 0x36 0xf0 0x82 0x08 0xc8 0x12 0xb6 0x49 0xf5 0xe0 0xea 0x37
0x56 0x9d 0xd6 0xa1 0xdc 0x8e 0x5e 0x10 0xca 0xa2 0xae 0x0d
Process finished with exit code 0
```

评测编号	提交时间	提交状态	代码语言	最大运行时间	最大运行内存	详细信息
16317	2022-04-22 00:48:08	Accepted	Python	2056ms	9004KB	

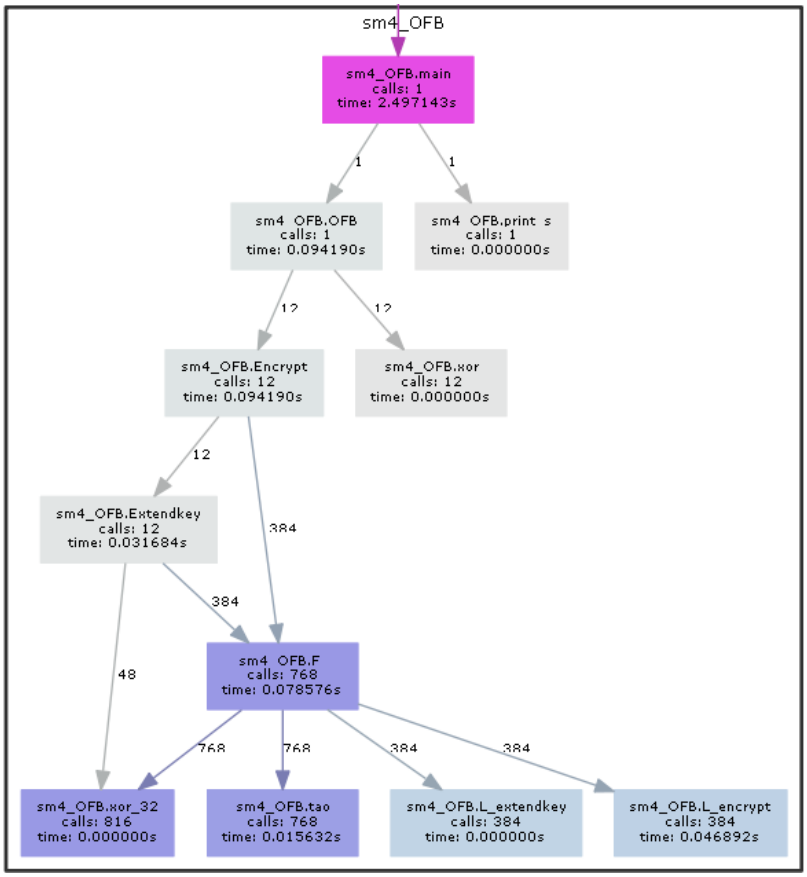
3、讨论与思考

- CFB——密文反馈（Cipher Feedback）
- CFB 模式先将反馈向量FB_i与密钥加密，与明文块P_i异或后得到密文块C_i，后更新反馈向量，左移n位并用C_i填充。
- CFB 模式对最后一组明/密文的处理同 CTR 模式。
- 由于 CFB 模式中分组密码是以流密码方式使用，所以加密和解密操作完全相同，因此无法适用于公钥密码系统，只能适用于对称密钥密码系统。

六、SM4 - OFB 模式

1、算法流程

函数调用关系图如下（以解密为例）：



2、测试样例及结果截图（结果过多，仅展示一组输出及 OJ 截图）

```
0xaf 0x88 0xbc 0x99 0xa7 0x97 0xb8 0x2e 0x7d 0xd8 0x92 0x44 0xcb 0x4a 0xbd 0x70
0xe8 0xa8 0x44 0x7b 0x35 0x4a 0xaf 0x0d 0x1a 0x33 0xdd 0x08 0x42 0xa0 0xee 0xc1
0x1a 0x17 0x8c 0xf8 0x1b 0xc1 0x2d 0x1f 0x3b 0xe7 0xb2 0x28 0xfa 0x13 0xfa 0x79
0xb2 0x92 0x50 0x6d 0x3f 0x22 0x86 0x56 0x94 0x78 0xe5 0x2e 0xb6 0x6f 0x03 0xc1
0xd8 0x81 0xb4 0xe0 0x45 0x28 0xe4 0x30 0x42 0xa5 0xc2 0x44 0x6c 0x06 0x6d 0xe3
0xfa 0x79
Process finished with exit code 0
```

评测编号	提交时间	提交状态	代码语言	最大运行时间	最大运行内存	详细信息
16318	2022-04-22 00:48:34	Accepted	Python	464ms	8736KB	

3、讨论与思考

- OFB——密文反馈（Output Feedback）
- OFB 模式先将反馈向量FB_i与密钥加密得到O_i，与明文块C_i异或后得到密文块P_i，后更新反馈向量，左移位并用O_i填充。
- OFB 模式对最后一组明/密文的处理同 CTR 模式。
- 密码算法的输出O_i会反馈到密码算法的输入中，OFB 模式并不是通过密码算法对明文直接加密，而是通过将明文分组和密码算法的输出进行$\oplus$来产生密文分组。

【收获与建议】

1、收获

- ~~加深了对 SM4 及 5 种工作模式的理解。~~
- ~~提高了对 Python 的熟练度。~~
- ~~巩固了排版能力。~~
- 收获了劳累与繁忙。

2、建议

- 无。

【思考题】

1、 请分析在使用 SM4 算法进行加密时，轮函数在各个阶段的运行时间占比。

- 对加密（enc）和密钥拓展（ext）中调用轮函数F（非线性变换τ和线性变换L）的系统时间进行统计（输入为 1 次加密的实例）：

	τ_ext	L_ext	τ_enc	L_enc
sum	1.94E-04	1.23E-04	1.63E-04	1.45E-04
average	6.07E-06	3.84E-06	5.10E-06	4.53E-06
sum[1:]	1.80E-04	1.14E-04	1.57E-04	1.39E-04
average[1:	5.79E-06	3.67E-06	5.07E-06	4.47E-06

	ext	enc
sum	3.17E-04	3.08E-04
average	9.91E-06	9.63E-06

	τ	L
sum	3.58E-04	2.68E-04
average	1.12E-05	8.37E-06

- 由于第一次调用时的时间比之后调用的时间均长，统计了第一次之后调用的时间。
- 由结果可以看出：①对于密钥拓展阶段和加密阶段，轮函数运算的运行时间基本相同；②轮函数中非线性变换τ和线性变换L运行时间占比大约为$4:3$。

2、 SM4 算法与 DES、AES 算法相比有什么异同。

同：均为对称密码中的**分组密码**，加解密都是对明文/密文进行分组后逐组进行加解密。分组密码的基本原理：

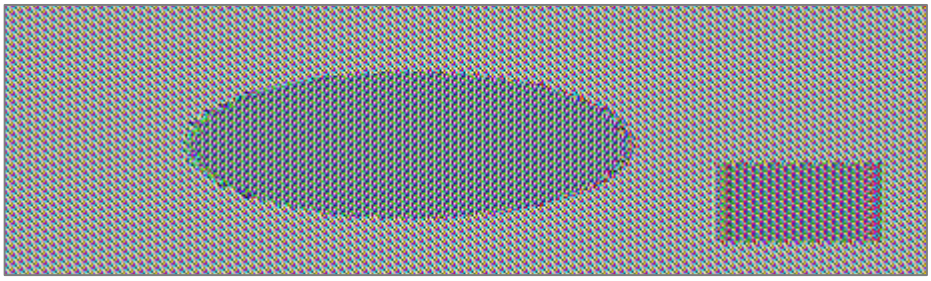
- 代换：明文分组到密文分组的可逆变换
- 扩散：将明文的统计特性散布到密文中去，使得明文的每一位影响密文中多位的值
- 混淆：使密文和密钥之间的统计关系变得尽可能复杂，使攻击者无法得到密钥，通常使用复杂的代换算法实现

异：

算法	密钥长度（bit）	分组长度（bit）	加/解密轮数	算法结构
DES	56	64	16	平衡 Feistel
AES	128/192/256	128	10/12/14	SPN
SM4	128	128	32	非平衡 Feistel

3、 （选做）分别使用 ECB 模式和 CBC 模式加密一个 bmp 图片，观察加密后的图片文件的特征，并结合工作模式的特点说明产生该现象的原因，只需要给出加密后的图片结果以及分析即可。（注：对于 bmp 图片文件，其前 54 字节包含了图片的头信息。因此，需要拼接原图片的前 54 字节与加密后图片第 55 字节开始的所有字节，构成一个合法的 bmp 图片文件）

ECB 加密后图片：



CBC 加密后图片：

分析：

- ECB 模式各分组间没有关系，相同分组加密后的结果相同，因此会暴漏明文的一些信息。如在 ECB 加密后的图片中能看出原图的形状特征。
- CBC 模式前面分组的结果会对后面的分组产生影响，保护了图片的结构信息，因此在 CBC 加密后的图片中很难分析出原图的形状特征。