SM3 rho 攻击

基本思路：

代码使用 python 中的库 gmmsl 中的 sm3 函数。首先对某个数据进行哈希，得到它的哈希值，将其存入一个列表中，在对哈希值进行哈希，这样不断哈希下去，直到某次哈希出来的结果在该列表中，此时攻击成功。

在代码中，为了防止死循环，我设置了一个最大值为 100000，即最多进行哈希操作 100000 次。超过则说明攻击失败。length 表示碰撞的长度，代表十六进制数。

结果：

16bits：

```
rho攻击成功
411
68b1
0.3089911937713623
```

20bits：

```
rho攻击成功
575
3b34a
0.47092294692993164
```

24bits：

```
rho攻击成功
7491
bd4a7e
5.649413824081421
```

28bits：

```
rho攻击成功
15385
9cf429f
12.24733018875122
```

32bits：

```
rho攻击成功
15385
9cf429f0
12.36038851737976
```

36bits：失败

```
rho攻击失败
138.26934003829956
```

运行截图：

由此可见随着碰撞的 bit 数增加，hash 次数随之增大，耗时增多，但也会有特殊情况，比如 28bits 与 32bits 时。