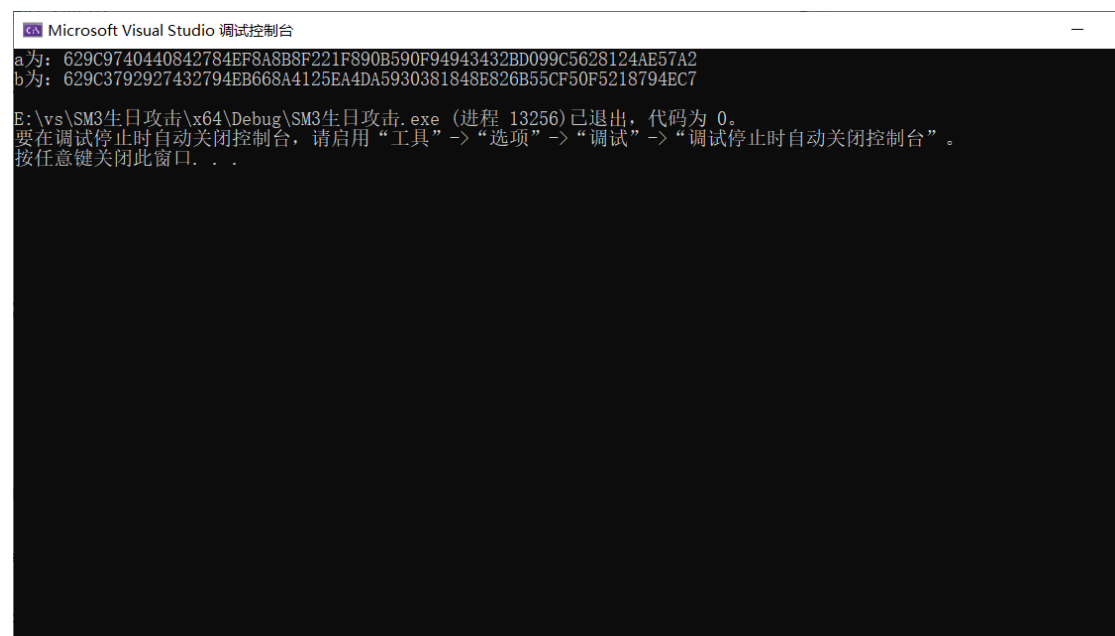


Rho 算法核心是要找到一个环，当两个字符串经 SM3 加密之后得到的结果相同时就发生了碰撞，我们可以利用 $a=SM3(a), b=SM3(SM3(b))$ ，由于 b 变化更快，所以我们总会找到一个环满足碰撞，本次实验选取前 16bit 和 20bit 相同即为发生碰撞。由于 SM3 生成结果为 16 进制字符串，为了统一我们将原始随机字符串也变为十六进制随机字符串，并利用输入为十六进制字符串的 SM_2 进行加密。

```
//Rho
string a = randomstr(8); //生成随机十六进制字符串
string b = SM3_2(a);
while (a.substr(0, 5) != b.substr(0, 5)) //找环，前5位相等即发生碰撞
{
    a = SM3_2(a);
    b = SM3_2(SM3_2(b));
}
cout << "a为: " << a << endl;
cout << "b为: " << b << endl;
```

当选取前 16bit 相同碰撞时，结果如下：



```
Microsoft Visual Studio 调试控制台
a为: 629C9740440842784EF8A8B8F221F890B590F94943432BD099C5628124AE57A2
b为: 629C9740440842784EF8A8B8F221F890B590F94943432BD099C5628124AE57A2
E:\vs\SM3生日攻击\x64\Debug\SM3生日攻击.exe (进程 13256) 已退出，代码为 0。
要在调试停止时自动关闭控制台，请启用“工具”->“选项”->“调试”->“调试停止时自动关闭控制台”。
按任意键关闭此窗口。 . . .
```

当选取前 20bit 相同碰撞时，结果如下：

```
Microsoft Visual Studio 调试控制台
a为: A1B6EAD824CB77D78BC9756F070D4361D334A0B7A2E5245C26B413F41924B563
b为: A1B6E81D76C78D77C6D2AB1013FED7F985CEF4F1B8BCB5009EA4B00D57540D8C

E:\vs\SM3生日攻击\x64\Debug\SM3生日攻击.exe (进程 22908) 已退出, 代码为 0。
要在调试停止时自动关闭控制台, 请启用“工具”->“选项”->“调试”->“调试停止时自动关闭控制台”。
按任意键关闭此窗口. . .
```