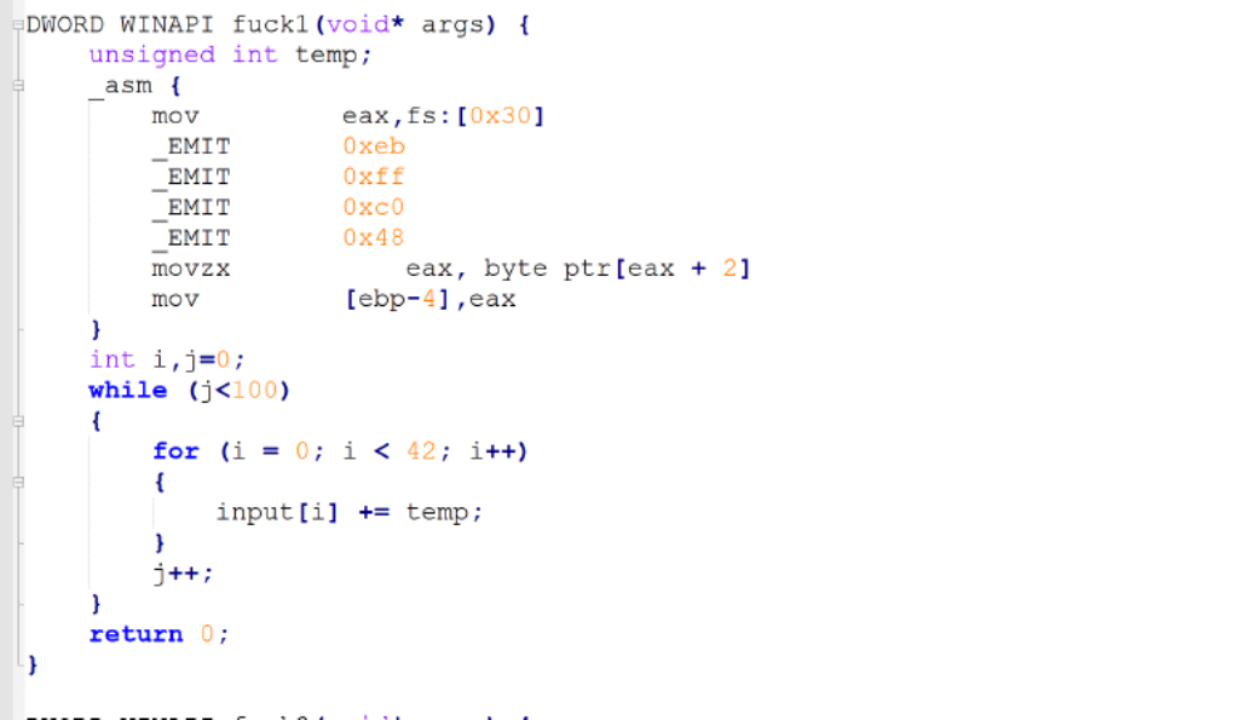
1.程序开了三个线程，第三个线程是个死循环，waitformultipleobjects只等待前俩个线程，第三个线程主要是疯狂检测调试，如下图所示：



2.花指令部分主要是指令重叠和利用call ret来实现的花真正加密操作只有移位加法和xor，如下图所示：



源码这边temp是个调试标记为非调试状态为0，暗桩罢了。

3.编写exp脚本如下所示：

```

a = ''

temp = [221, 91, 158, 29, 32, 158, 144, 145, 144, 144, 145, 146, 222, 139, 17,

209, 30, 158, 139, 81, 17, 80, 81, 139, 158, 93, 93, 17, 139, 144, 18, 145, 80,

18, 210, 145, 146, 30, 158, 144, 210, 159]

for i in range(len(temp)):

temp[i] -= 0x23

temp[i] &= 0xff

temp[i] ^= 0x23

temp[i] = ((temp[i] << 2) ^ (temp[i] >> 6)) & 0xff

a += chr(temp[i])

print a

#flag{a959951b-76ca-4784-add7-93583251ca92}

```

运行脚本结果，如下图所示：



则题目答案为flag{a959951b-76ca-4784-add7-93583251ca92}。