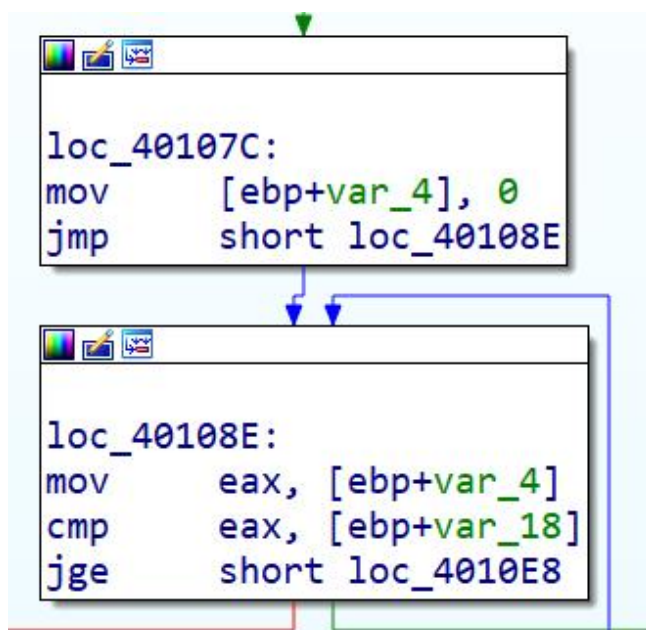


1. 第一个比较，确定 flag 长度为 8，其中 `ebp + Str` 处存储输入字符串，`ebp + var_18` 处存储输入字符串的长度；

```
push    offset aPlaseGiveMeYou ; "Plase give me your answer:\n"
call    _printf
add     esp, 4
lea     eax, [ebp+Str]
push    eax
push    offset Format          ; "%s"
call    _scanf
add     esp, 8
lea     ecx, [ebp+Str]
push    ecx                    ; Str
call    _strlen
add     esp, 4
mov     [ebp+var_18], eax
cmp     [ebp+var_18], 8
jz      short loc_40107C
```

2. 长度确定为 8 后，进入 for 循环，其中 `ebp + var_4` 处存储循环计数器（后续解答中都称为 i）；

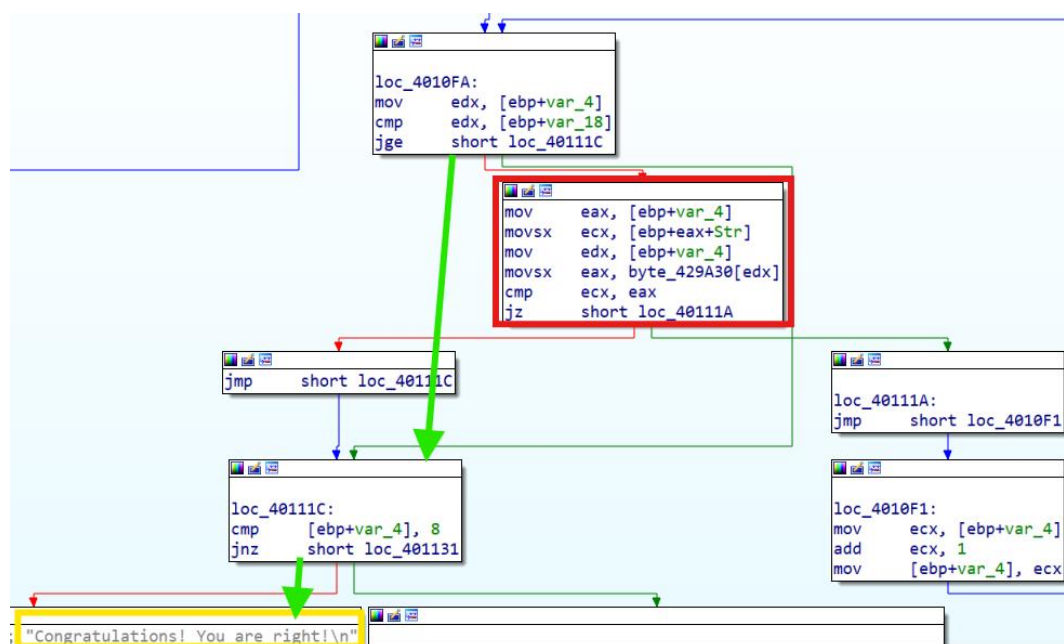


3. 核心计算逻辑如下：依次取出输入的字符串的每个字符，首先分别和 '7Eh' 做与操作、和 '80h' 做与操作并右移 7 位、和 '1' 做与操作并左移 7 位，然后三个结果做或操作，然后获得的结果与 `byte_429A38` 对应位置的值做异或操作，遍历 8 个字符后，退出循环，其物理含义为交换 1 字节中最高和最低两

个 bit 位;

```
mov     ecx, [ebp+var_4]
movsx   edx, [ebp+ecx+Str]
and     edx, 7Eh
mov     eax, [ebp+var_4]
movsx   ecx, [ebp+eax+Str]
and     ecx, 80h
sar     ecx, 7
or      edx, ecx
mov     eax, [ebp+var_4]
movsx   ecx, [ebp+eax+Str]
and     ecx, 1
shl     ecx, 7
or      edx, ecx
mov     eax, [ebp+var_4]
mov     [ebp+eax+Str], dl
mov     ecx, [ebp+var_4]
movsx   edx, [ebp+ecx+Str]
mov     eax, [ebp+var_4]
movsx   ecx, byte_429A38[eax]
xor     edx, ecx
mov     eax, [ebp+var_4]
mov     [ebp+eax+Str], dl
jmp     short loc_401085
```

4. 将 i 置零后, 再次遍历字符串进行判断, 与 byte\_429A30 进行比较 (红框内), 若相等, 则输出“Congratulations!”, 为目标的成功点, 成功路线为绿色箭头所指路径;

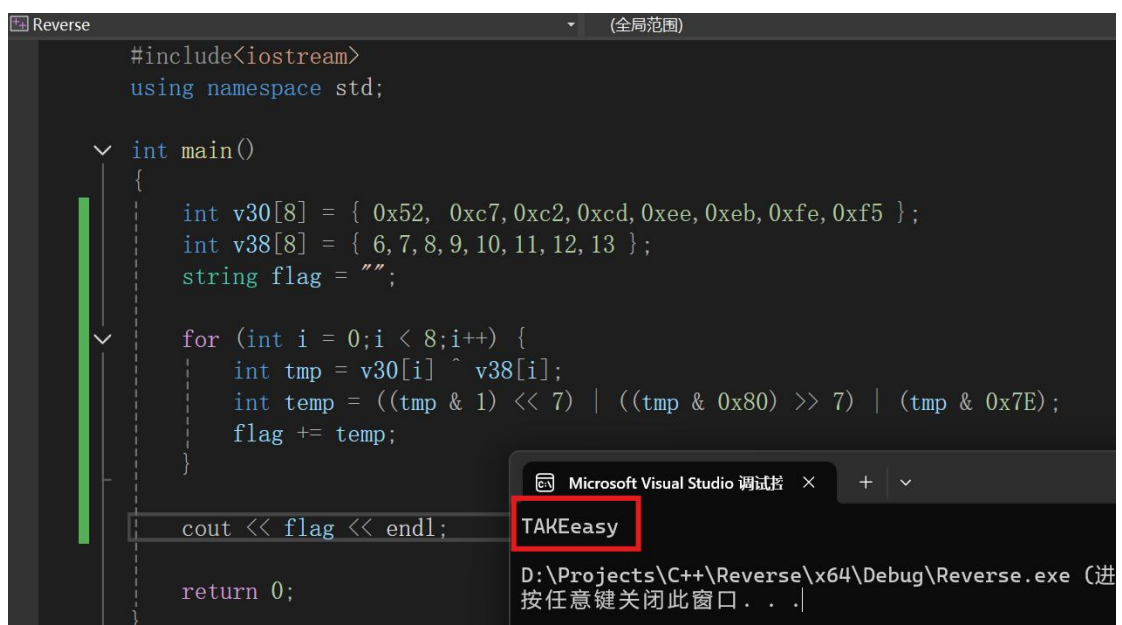


5. 逻辑明确后，去找两个用到的数组，如下图（仅截取使用到的部分）：

unk_429A30	db 52h	unk_429A38	db 6
	db 0C7h		db 7
	db 0C2h		db 8
	db 0CDh		db 9
	db 0EEh		db 0Ah
	db 0EBh		db 0Bh
	db 0FEh		db 0Ch
	db 0F5h		db 0Dh

; BYTE byte 429A38[24]

6. 最后写出逆向程序，获得 flag：TAKEeasy。



```
#include<iostream>
using namespace std;

int main()
{
    int v30[8] = { 0x52, 0xc7, 0xc2, 0xcd, 0xee, 0xeb, 0xfe, 0xf5 };
    int v38[8] = { 6, 7, 8, 9, 10, 11, 12, 13 };
    string flag = "";

    for (int i = 0; i < 8; i++) {
        int tmp = v30[i] ^ v38[i];
        int temp = ((tmp & 1) << 7) | ((tmp & 0x80) >> 7) | (tmp & 0x7E);
        flag += temp;
    }

    cout << flag << endl;

    return 0;
}
```

Microsoft Visual Studio 调试器

TAKEeasy

D:\Projects\C++\Reverse\x64\Debug\Reverse.exe (进  
按任意键关闭此窗口. . .)