

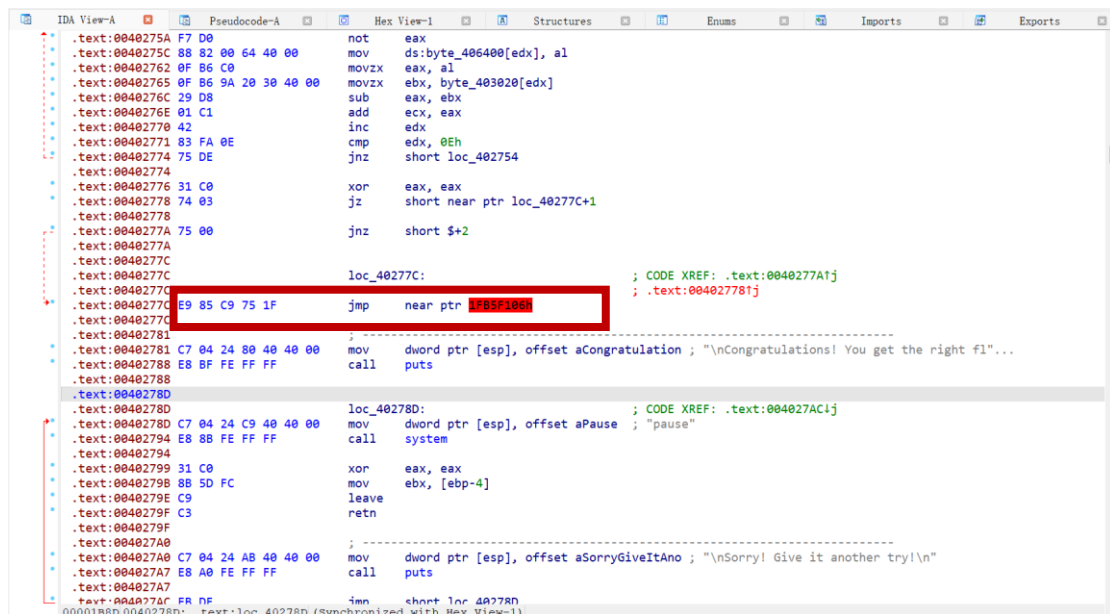
# Homework-3 Report

姓名：项 枫      学号：2022211570

## 一、解题过程

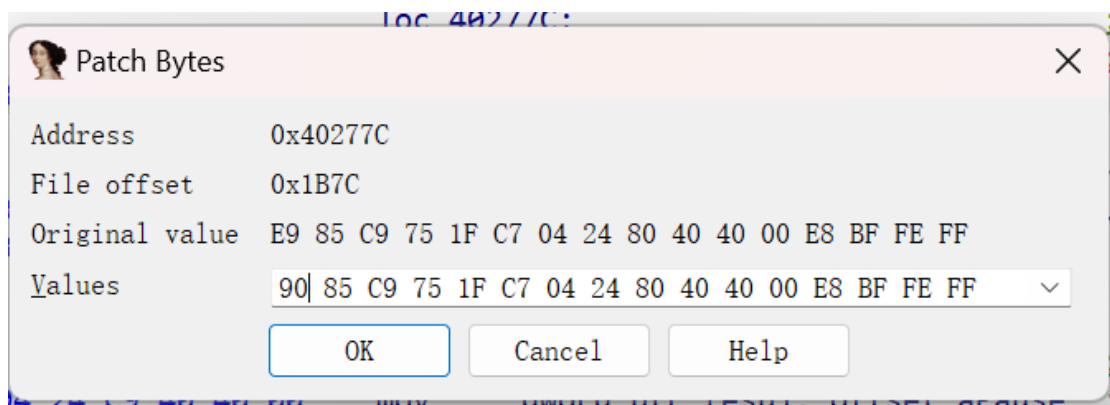
### 1、花指令解决

#### (1) 存在 jmp 花指令



```
.text:0040275A F7 D0      not     eax
.text:0040275C 88 82 00 64 40 00    mov     ds:byte_406400[edx], al
.text:00402762 86 C0      movzx   eax, al
.text:00402765 0F 86 9A 20 30 40 00    movzx   ebx, byte_403020[edx]
.text:0040276C 29 D8      sub     eax, ebx
.text:0040276E 01 C1      add     ecx, eax
.text:00402770 42        inc     edx
.text:00402771 83 FA 0E      cmp     edx, 0Eh
.text:00402774 75 DE      jnz     short loc_402754
.text:00402774
.text:00402776 31 C0      xor     eax, eax
.text:00402778 74 03      jz      short near ptr loc_40277C+1
.text:0040277A 75 00      jnz     short $+2
.text:0040277A
.text:0040277C      loc_40277C:                                ; CODE XREF: .text:0040277A1j
.text:0040277C      jmp     near ptr 1F85F100                ; .text:004027781j
.text:0040277C
.text:00402781      ;
.text:00402781 C7 04 24 C9 40 40 00    mov     dword ptr [esp], offset aCongratulations ; "\nCongratulations! You get the right f1"...
.text:00402788 E8 BF FE FF FF      call    puts
.text:00402788
.text:0040278D      loc_40278D:                                ; CODE XREF: .text:004027AC1j
.text:0040278D C7 04 24 C9 40 40 00    mov     dword ptr [esp], offset aPause ; "pause"
.text:00402794 E8 8B FE FF FF      call    system
.text:00402794
.text:00402799 31 C0      xor     eax, eax
.text:0040279B 8B 5D FC      mov     ebx, [ebp-4]
.text:0040279E C9        leave  eax
.text:0040279F C3        retn
.text:0040279F
.text:004027A0      ;
.text:004027A0 C7 04 24 AB 40 40 00    mov     dword ptr [esp], offset aSorryGiveItAno ; "\nSorry! Give it another try!\n"
.text:004027A7 E8 A0 FE FF FF      call    puts
.text:004027A7
.text:004027A7 FR NF      imn     short loc_40278D
00001B8D:0040278D: .text:loc_40278D (synchronized with Hex View-1)
```

#### (2) 将 E9 改成 90



#### (3) F5 反编译

```

1 int __cdecl main(int argc, const char **argv, const char **envp)
2 {
3     int v3; // edx
4     int v4; // ecx
5     unsigned __int8 v5; // al
6
7     sub_401720();
8     puts("----- This one is pretty easy! JUST ENTER YOUR FLAG: ");
9     scanf("%s", byte_406400);
10    v3 = 0;
11    v4 = 0;
12    do
13    {
14        v5 = ~byte_406400[v3];
15        byte_406400[v3] = v5;
16        v4 += v5 - (unsigned __int8)byte_403020[v3++];
17    }
18    while ( v3 != 14 );
19    if ( v4 )
20        puts("\nSorry! Give it another try!\n");
21    else
22        puts("\nCongratulations! You get the right flag!\n");
23    system("pause");
24    return 0;
25 }

```

## 2、main 函数分析

(1) 12—17 行：将输入字符串先逐字符取反，再与 byte\_403020 处字符串逐个字符进行相减，相减结果进行相加存入 v4。

(2) 19—22 行：只有 v4 为 0 时，才输出"\nCongratulations! You get the right flag!\n"。

综上，循环 14 次，可知字符串长度为 14；flag 为 byte\_403020 处字符串逐字符取反。

## 3、解 flag 程序

(1) 程序代码 (C++) 如下：

```

#include<bits/stdc++.h>

int main(){

    char flag[]={0xBD,0x9A,0x9E,0x8B,0xD5,0xCF,0x92,
                  0x96,0x9C,0x8D,0x90,0x91,0xD5,0xDE};
}

```

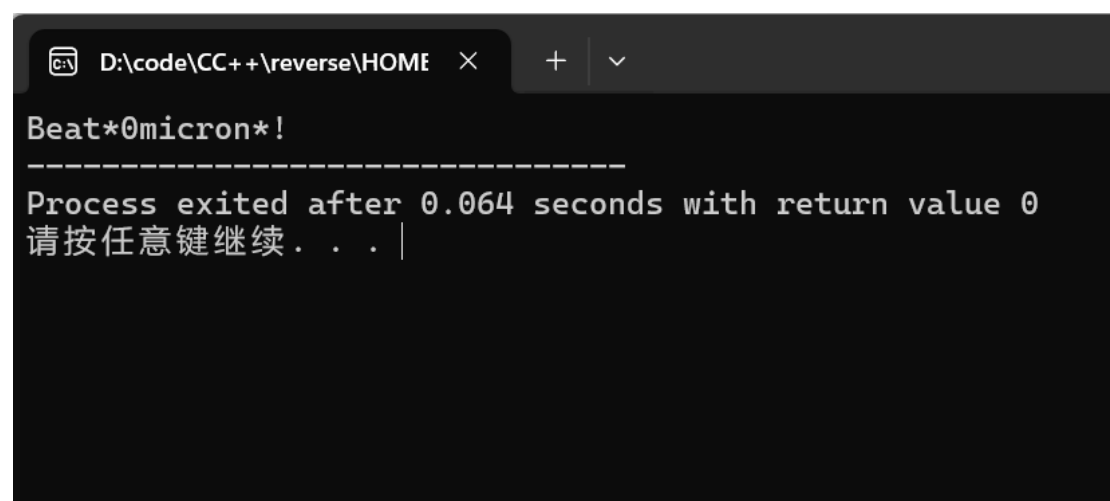
```
int i;

for(i=0;i<strlen(flag);i++){
    flag[i]=~flag[i];
}

printf("%s",flag);

return 0;
}
```

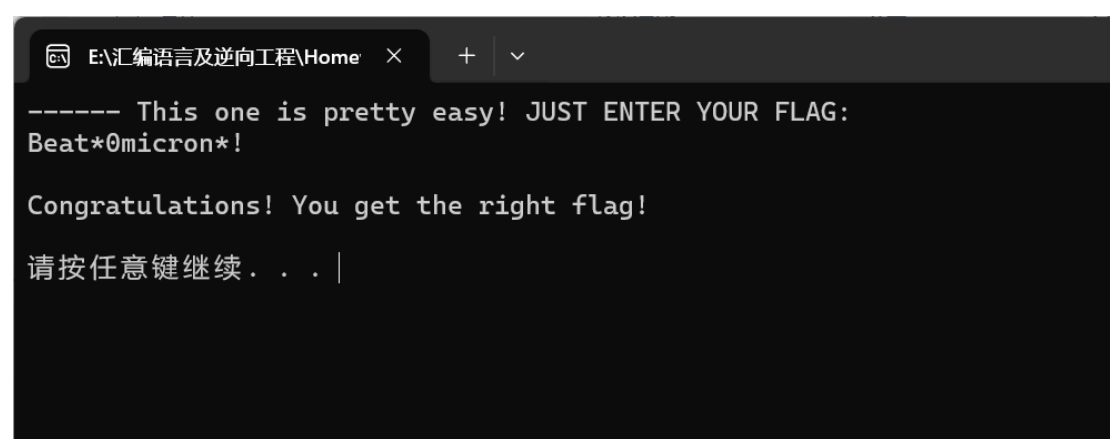
## (2) 程序运行结果



```
D:\code\CC++\reverse\HOME × + v

Beat*0micron*!
-----
Process exited after 0.064 seconds with return value 0
请按任意键继续 . . . |
```

## 4、flag 成功通过



```
E:\汇编语言及逆向工程\Home × + v

----- This one is pretty easy! JUST ENTER YOUR FLAG:
Beat*0micron*!

Congratulations! You get the right flag!

请按任意键继续 . . . |
```

## 二、收获和感受

通过本次作业，初步了解了花指令及其逆向的操作。

最后，感谢潘老师课上辛勤的教学。