

```

11 v10 = __readfsqword(0x28u);
12 v5 = 0LL;
13 v6 = 0;
14 v7 = 0LL;
15 v8 = 0;
16 v9 = 0;
17 __isoc99_scanf("%s", &v5);
18 if ( (unsigned int)sub_4006D6((const char *)&v5) )// v5长度为10. 0~4为可输入字符
19 {
20     v4 = sub_400758(&v5, 0LL, 10LL);           // 输入的二叉树结构
21     sub_400807(v4, &v7);                       // 中序遍历改变输入顺序
22     v9 = 0;
23     sub_400881(&v7);                           // 按一定顺序插入输入的值
24     if ( (unsigned int)sub_400917() )
25     {
26         puts("TQL!");
27         printf("flag{");
28         printf("%s", (const char *)&v5);
29         puts("}");
30     }
31 }

```

首先读入，（0-4）为可读入内容得到v5  
 然后按二叉树的方式顺序存储长度为10的输入得到v4  
 然后根据中序遍历改变输入的顺序得到v7  
 用v7按顺序填充内容  
 是个5x5的数独  
 可以得到

```

0x6012a0 <x+544>: 0x78 0x12 0x60 0x0 0x0 0x
gdb-peda$ q
root@kali:/home/kali/Desktop/ctf# chmod 777 number_game
root@kali:/home/kali/Desktop/ctf# ./number_game
1134240024
TQL!
flag{1134240024}

```

flag{1134240024}