c

南開大學

汇编语言与逆向技术课程实验报告

实验十： CTF（Capture The Flag）夺旗赛

学　　院　网络空间安全学院

专　　业　信息安全、法学双学位

学　　号　2212000

姓　　名　宋奕纬

班　　级　1061

一、实验目的

1、 进一步熟悉静态反汇编工具 IDA Freeware;

2、 掌握对二进制代码内部逻辑关系的分析;

3、 掌握对二进制代码的修改和保存。

二、实验原理

1、 **实验环境：**反汇编工具 IDA Freeware

2、 **实验原理：**通过 IDA 得到 game.exe 二进制代码的反汇编代码，利用汇编所学知识对反汇编代码的数学计算、数据结构、条件判断、分支结构进行识别与分析,并修改相关数据与指令，实现游戏的通关。

3、**实验目的：**分析 game.exe 二进制代码的主要逻辑结构与重要数据，修改其代码实现游戏通关，夺得 flag。

4、**相关信息：**

（1）CTF

CTF 是一种流行的信息安全竞赛形式，可意译为“夺旗赛”。其大致流程是，参赛团队之间通过进行攻防对抗、程序分析等形式，率先从主办方给出的比赛环境中得到一串具有一定格式的字符串或其他内容，并将其提交给主办方，从而夺得分数。CTF 竞赛模式具体分为以下三类：

一、解题模式（Jeopardy）

在解题模式 CTF 赛制中，参赛队伍可以通过互联网或者现场网络参与，这种模式的 CTF 竞赛与 ACM 编程竞赛、信息学奥赛比较类似，以解决网络安全技术挑战题目的分值和时间来排名，通常用于在线选拔赛。题目主要包含**逆向分析**、漏洞挖掘与利用、Web 渗透、密码、取证、隐写、安全编程等类别。

二、攻防模式（Attack-Defense）

在攻防模式 CTF 赛制中，参赛队伍在网络空间互相进行攻击和防守，挖掘网络服务漏洞并攻击对手服务来得分，修补自身服务漏洞进行防御来避免丢分。

三、混合模式（Mix）

结合了解题模式与攻防模式的 CTF 赛制，比如参赛队伍通过解题可以获取一些初始分数，然后通过攻防对抗进行得分增减的零和游戏，最终以得分高低分出胜负。

（2）解题

Flag 隐藏在 game.exe 的二进制代码中。通过对 game.exe 的修改，使 game.exe 能够顺利的执行，完成对 Flag 的解密。

可以通过 IDA Pro 修改静态资源与汇编指令，实现我们想要的功能。

三、实验过程

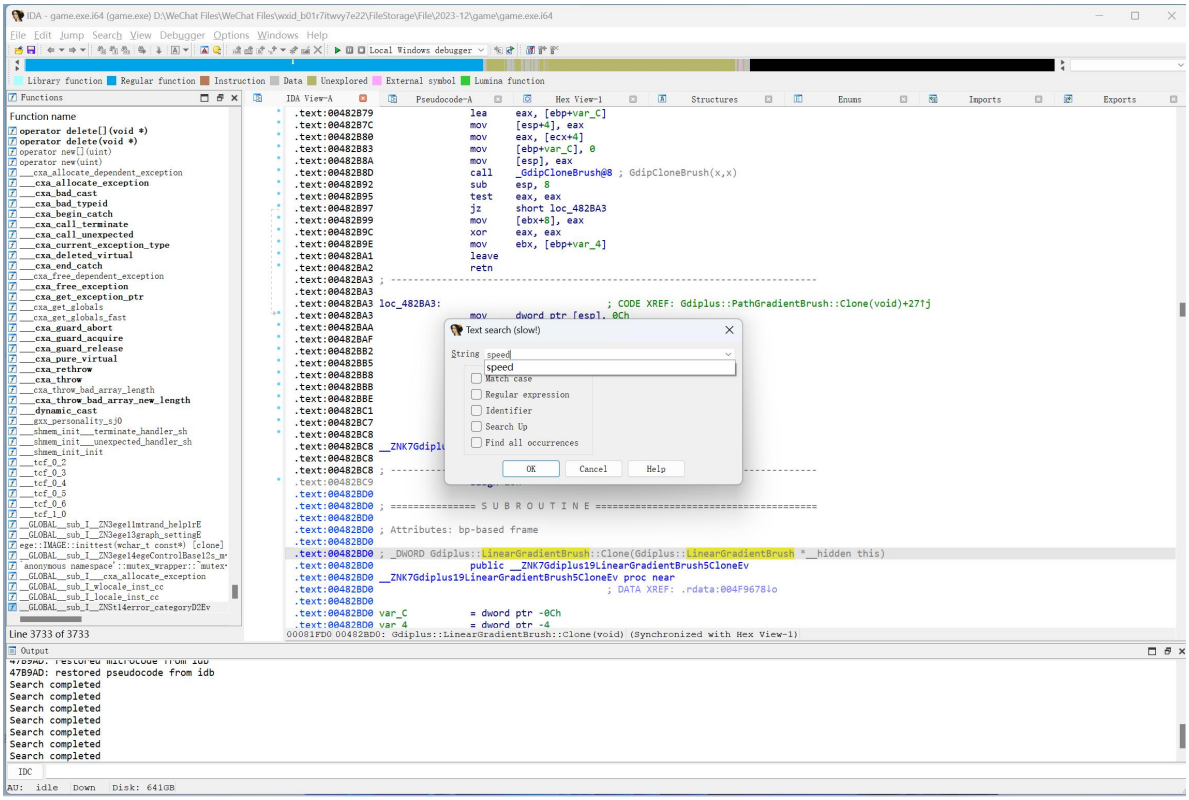
（一）逆向分析 game.exe 二进制代码的主要逻辑结构和重要数据。

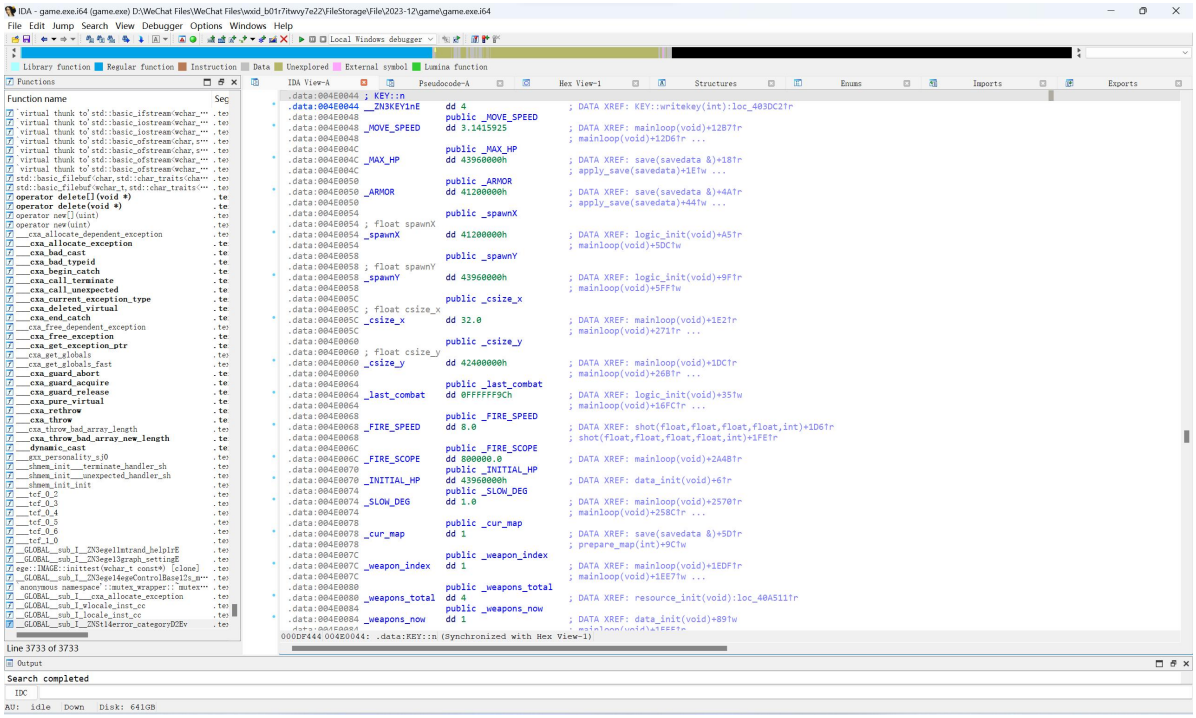
1、 **使用 IDA Freeware 打开 game.exe 文件，查看其二进制代码的反汇编代码。（先运行程序大概了解其基本情况）**

经过对游戏的测试，发现游戏人物首先会触发教学判定，然后进入正式关卡。在此过程中发现两点问题：其一，角色属性不够强，在游戏过程中容易死亡；其二，通过躲避与卡位刷完全部的怪物并且触发所有的 npc 对话后仍无法通关。故从两个方向进行研究与分析——其一加强角色属性（血量、攻击力、蓝量等），其二是寻找“拔旗”、使游戏通关、取得 flag 的条件。

（1）强化游戏角色

从以下几个方向考虑对角色的强化：移动速度、血量、蓝量、攻击力，关于血量又有两点思考——提升其上限与初始值使人物不容易死去、删掉其死亡的判定使其不死。但考虑到游戏代码往往牵一发而动全身，如果寻找到死亡的分支进行删除便可能导致整个游戏崩溃——故对角色属性进行修改时应从其静态资源修改进行考虑。（由上课时讲到的 Speed 属性受到启发，用 ALT+t 键进入搜索模式，寻找其属性对应的静态资源的位置）

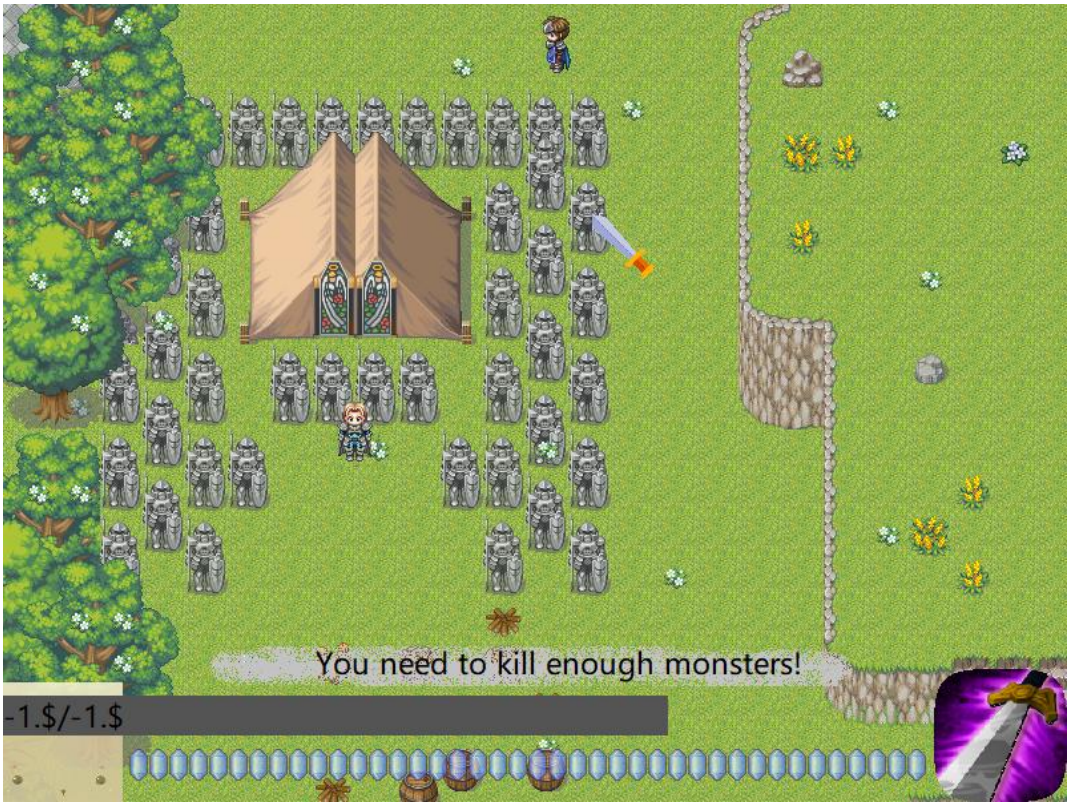




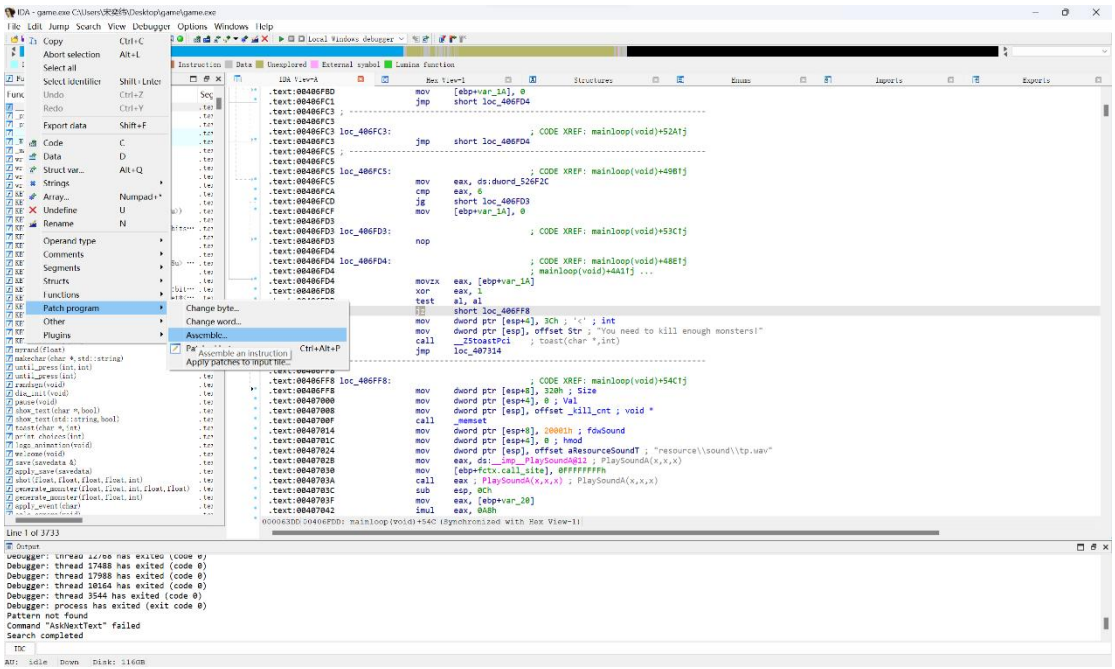
于是我们找到了其属性的静态资源（修改思路与过程在下一步给出）

（2）寻找触发游戏通关的条件：

在游戏中探索，出现以下界面：



发现到此处时未触发过关条件，无法进入下一关，针对该提示语进行搜索（用 alt+T 搜索“enough”即可找到）



发现此处为判断后的有条件跳转，猜测在此处其可以直接进行跳转，进而

可以实现无条件、不用战斗通关。

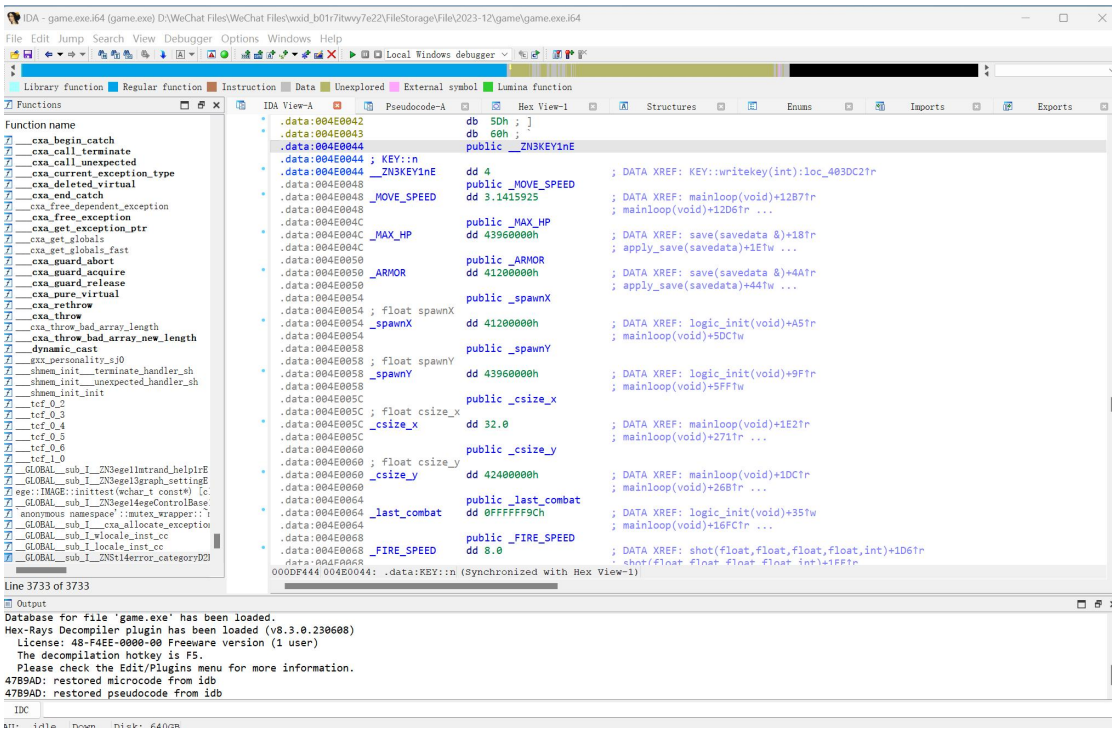
2、 使用 IDA 对静态资源与指令进行修改

（1）对静态资源的修改（强化属性）

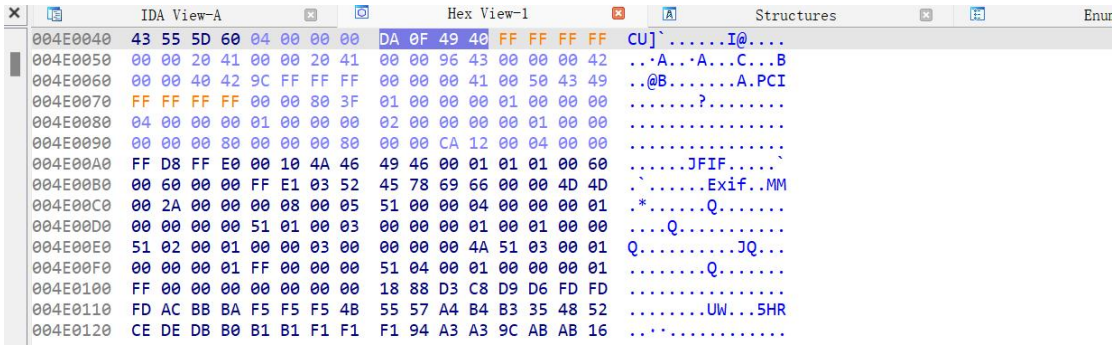
猜测其死亡判定为血量变成 0，故产生一个大胆的想法——如果一开始就让其血量为负数是否就可以实现角色永远不触发死亡判定。

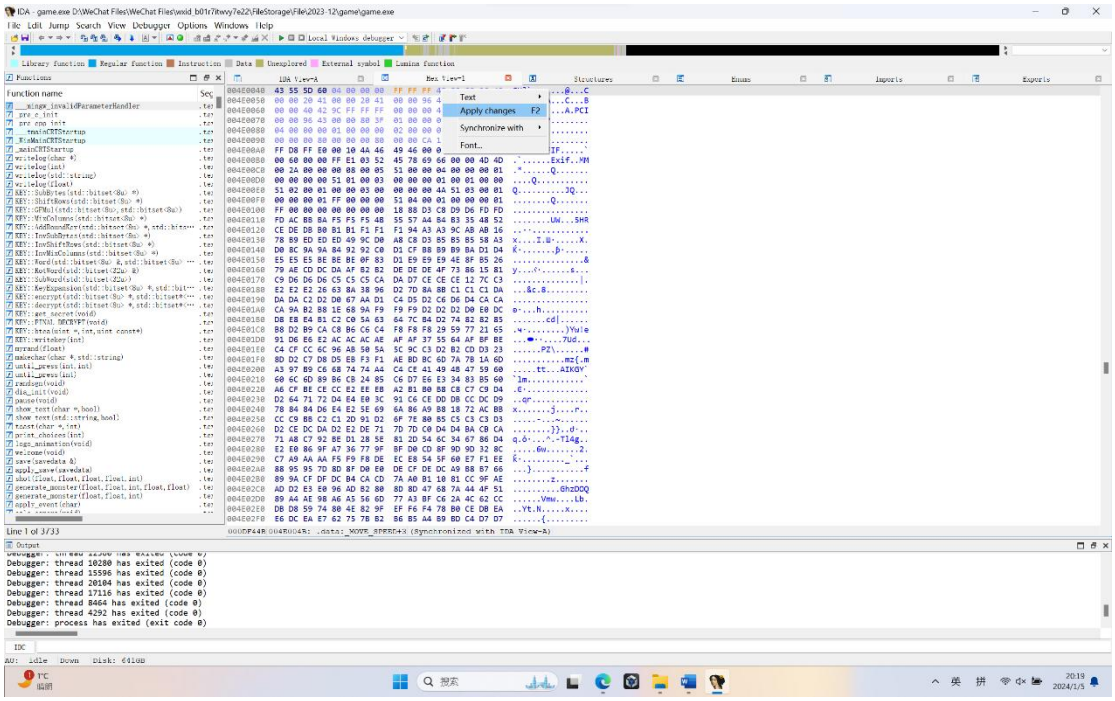
故主要对其两方面的属性进行修改——移动速度、血量。

①定位

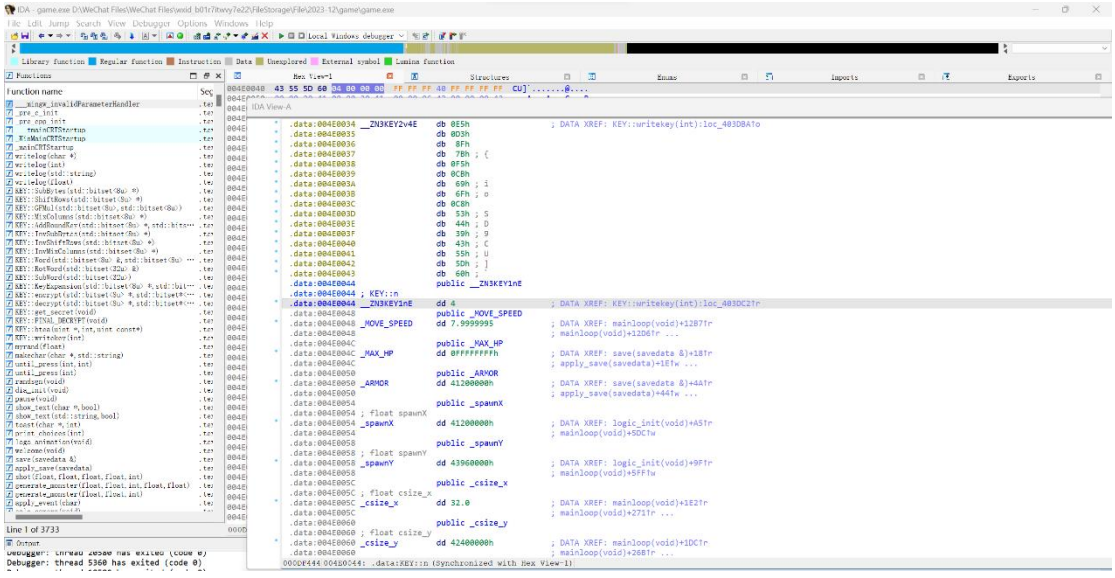


②在十六进制视图中（Hex View）找到指定区域，右键选择 Edit 对资源进行修改。修改完毕后，右键选择 Apply changes 应用修改。





③返回查看修改效果，观察到速度大大提升（此时速度过快，导致在游戏过程中会卡墙，故又将其适当调小），最大血量（_MAX_HP）和初始血量（_INITIAL_HP）均被设置成-1。



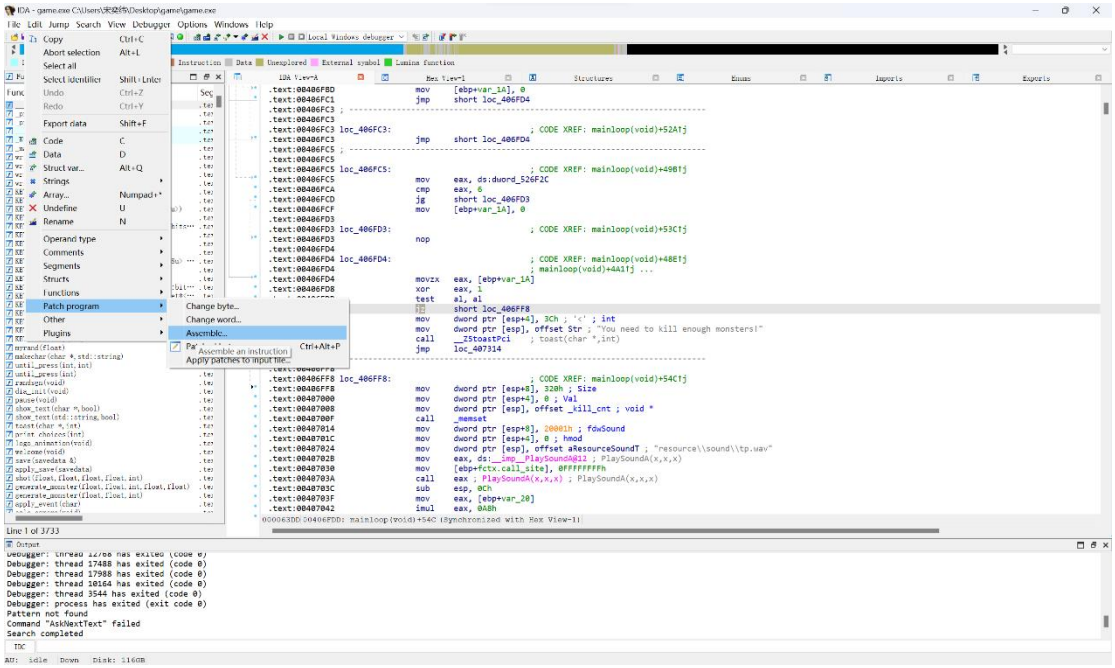
（2）对指令的修改（实现通关）

①定位

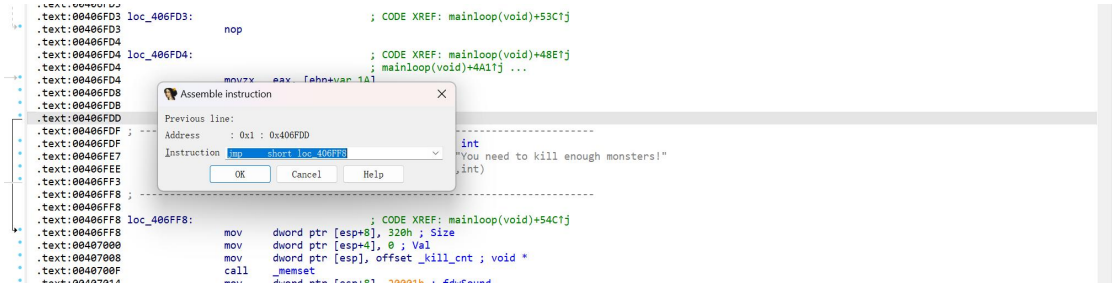
找到对应的“You need to kill enough monsters！”，发现该语句前有这样的指令：

```
test    al, al
jz      short loc_406FF8
```

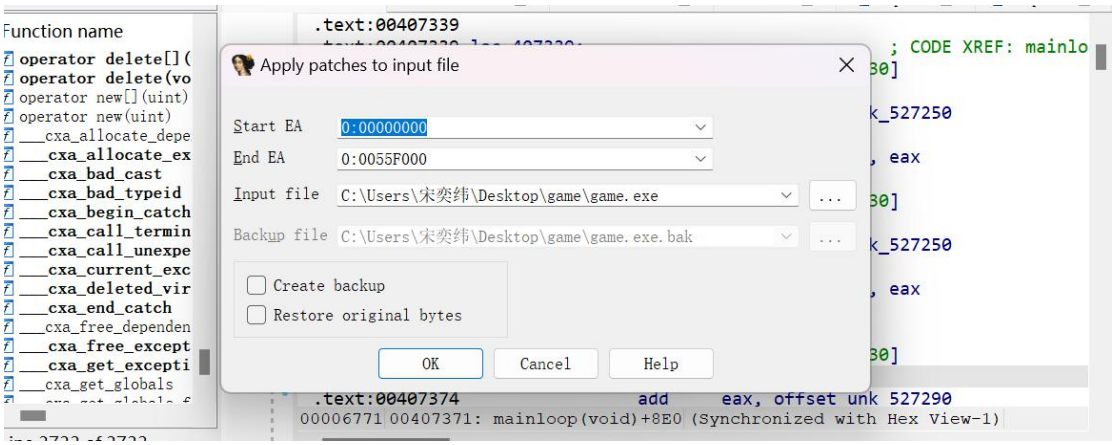
其中 jz 通过 ZF 标志位是否跳转，当执行到 JZ 指令时，如果 ZF=1 则跳转，如果 ZF=0，不跳转；之前条件过于繁杂，故直接将其改为 jmp，无条件跳转。



②点击 Edit->Patch program->Assemble，输入新的汇编指令。

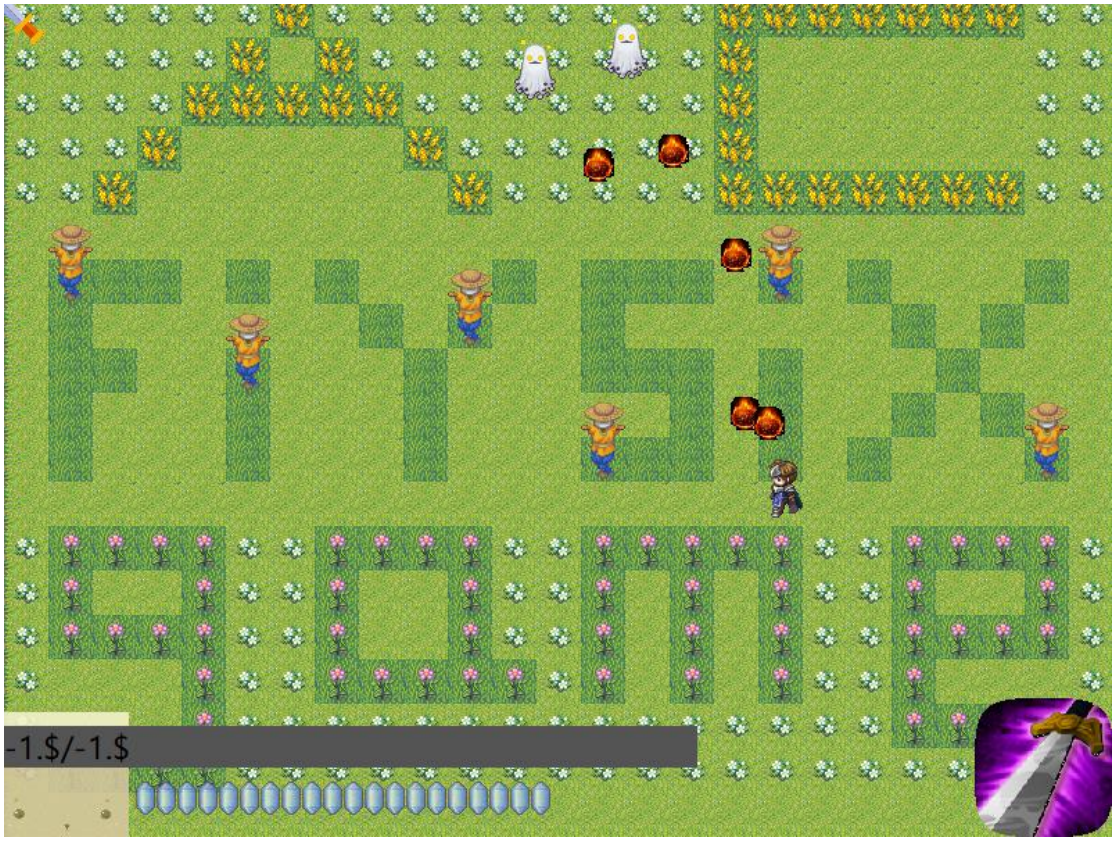


（3）保存

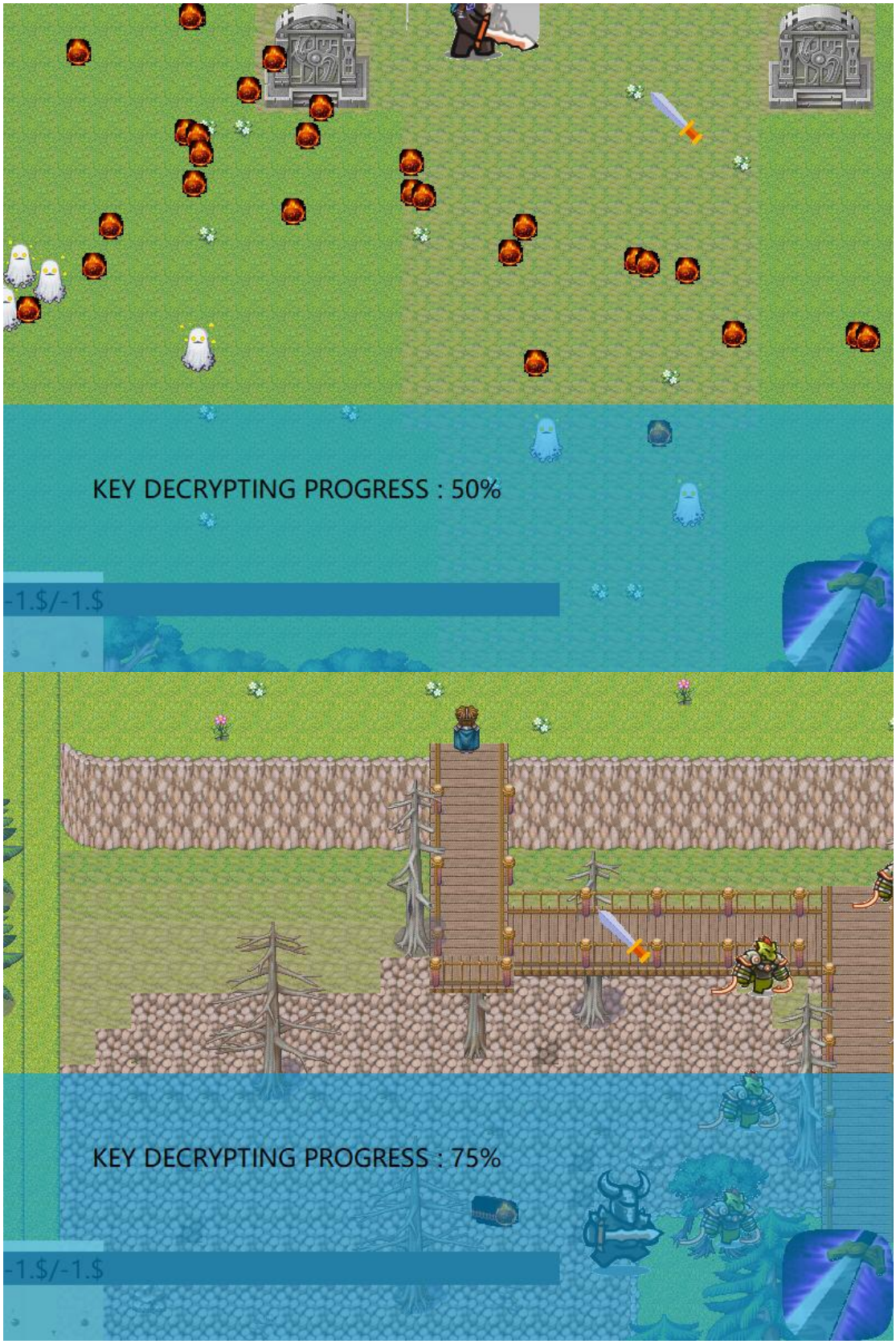


3、 进行游戏

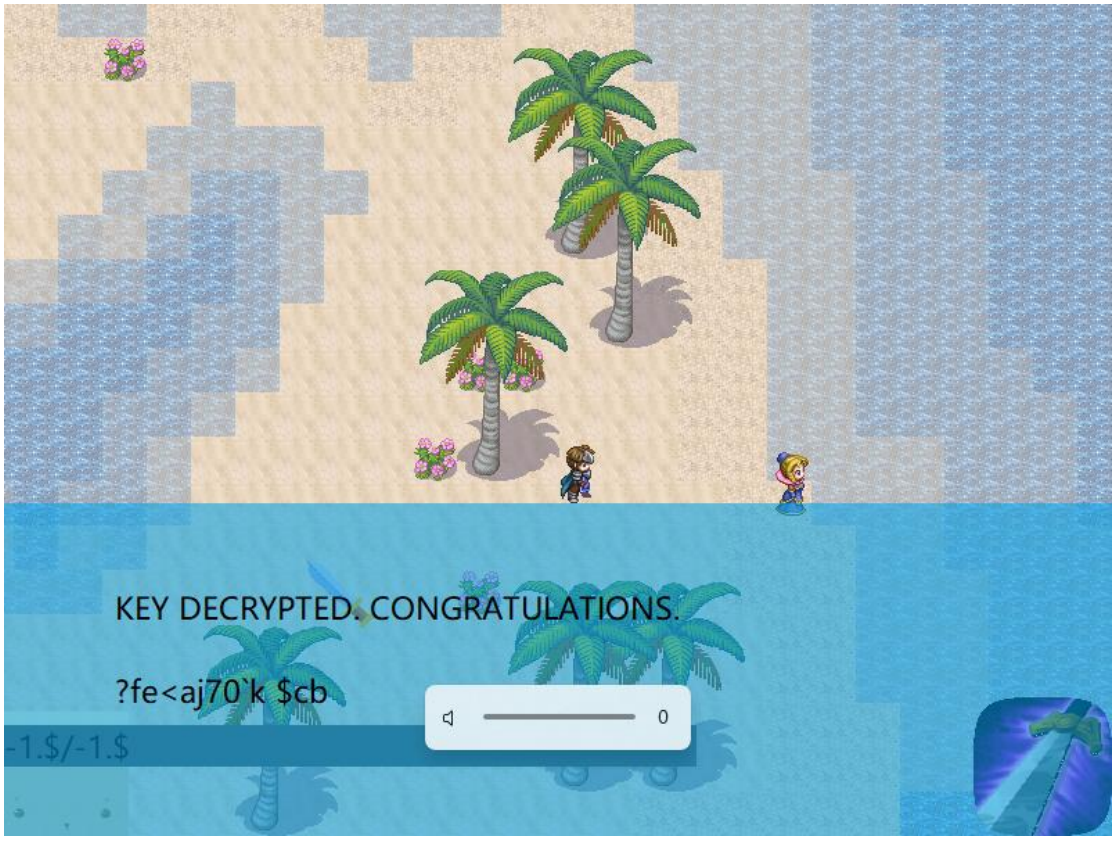
①进入游戏，发现速度极快，而且无法死亡



②不攻击怪物，直接前往下一关判定处



③进入爱琴海，通关游戏，找到 flag



四、实验结论及心得体会

1、 熟悉静态反汇编工具 IDA Freeware。

2、 熟悉反汇编代码的逆向分析过程。

3、 初步掌握了对较大程序的逆向分析、修改的能力。

4、 掌握反汇编语言中的数学计算、数据结构、条件判断、分支结构的识别和逆向分析。

5、 对地址、变量、指针都有了更深入的理解

6、 对汇编语言有了更深入的了解。