# C语言项目报告（14组）：魔塔游戏

## 摘要

“魔塔”创始于2005年，是一款经典的RPG策略游戏，即使是游戏琳琅满目的今天在国内仍有上万的魔塔发烧友。经过历代优化，它迄今为止已有上千种各具特色的版本。比如网页版魔塔（V2.3）允许玩家不用下载即可与好友在线畅玩。我们期望我们可以用C语言再现其中的函数，实现基本操作功能（我们必要时也采用C++编写代码）。同时，我们加入了一些自己的创意，比如新的魔塔地图。期望我们的作品能够给大家回味经典游戏的快乐。

## 1.介绍 & 问题声明

1.1Introduction

限于水平和精力，在本工程中我们设计了21层的基础魔塔，囊括了魔塔的游戏设定中核心的元素，勇士与怪的对战、与NPC互动的剧情以及亲切的道具设置。机关方面，我们将一个特殊的机关门的钥匙放到更高层的怪物包围圈中，以控制玩家后期才能挖掘到机关门内的资源，促进游戏的平衡性。原计划打算设置在指定位置才能触发的暗门机关，但出于实现难度和玩家体验的考虑，那种玄机塔式的复杂机关为我们所摒弃。剧情方面，即勇士被魔王暗算武功尽废，他从零开始层层挑战魔塔，路途中获得了贵人相助恢复了一些实力，战胜拦路小怪以及真假boss魔王最终成功营救被魔王掳走的公主。道具设置方面，我们参考了正版游戏的道具摆放，并讨论确定出钥匙、魔法药水等道具合理的铺置位置和对应数量。经过小组合作开发，最终完成了本工程的二十一层魔塔。

1.2Problem statement

1.2.0新地图的绘制和楼层间的过渡（梯子）、美化UI界面。

1.2.1英雄与NPC对话：如何触发、继续和结束？怎样解决台词切换时对话框闪烁的问题？

1.2.2魔法道具的注释:对于从未接触过魔塔类游戏的新手，如何告知该玩家魔法道具的作用，从而帮助玩家合理规划闯关路线或自己英雄的属性值？能不能实现当鼠标移动至道具位置便显示该道具的信息？类似地，能不能做出怪物图鉴，以方便玩家量力而行？

1.2.3英雄vs怪物: 魔塔游戏中对战环节是重中之重，对战的算法如何设计?网络上各式各样的计算方法参考中我们应该采取哪一种?如果都不合适，需要自己设计。另外由于21层魔塔比较基础，怪物实力水平“可望可及”，为英雄加入“暴击”是否妥当，影响游戏平衡么？

1.2.4胜负的判定：当玩家击败boss怪物，应视为游戏胜利。但是，应该当战胜boss之时即播放剧情动画退出游戏还是不立即退出，任由玩家回味探索，待玩家自行Esc退出？

1.2.5键盘与鼠标的游戏控制

1.3小组分工

|  |  |  |  |
| --- | --- | --- | --- |
| 姓名 | 黄林志 | 陈广兴 | 张潇文 |
| 任务 | 工程结构分析  算法转化 | 地图的准备和实现  调试 | 项目优化  代码运行时间调整 |

## 2.分析

分析魔塔的游戏特点，落实到代码编写上时，我们可以分成三个主要阶段完成。第一阶段资源的准备、地图的构建及道具摆放：为了让玩家能够切实地体会到回味经典游戏的满足感，提升游戏的还原度，需要前期在网上搜集魔塔的资源图片（英雄、道具、怪、障碍；等等）并进行编号命名，以利于二维矩阵导入。同时，制作txt对话文本。在二十一张的地图制作环节，可以考虑利用一些工具如“easyX”减轻工作负担，参考正版魔塔游戏的地图与道具摆放，增加设计的合理性。

第二阶段，各个功能对应函数头文件的制作，将各个算法的代码编写分配给小组成员，完成各自部分。重点函数合作完成，比如利用键盘钩子和鼠标控制人物或菜单光标的移动、对战板块、商店板块、互动对话、楼层间的梯子等等。

第三阶段，主函数编写（要定义所有窗口区域）与代码整合。将分工的任务整合到一个工程中，删节一些冗余代码。随后转入调试与修改环节。

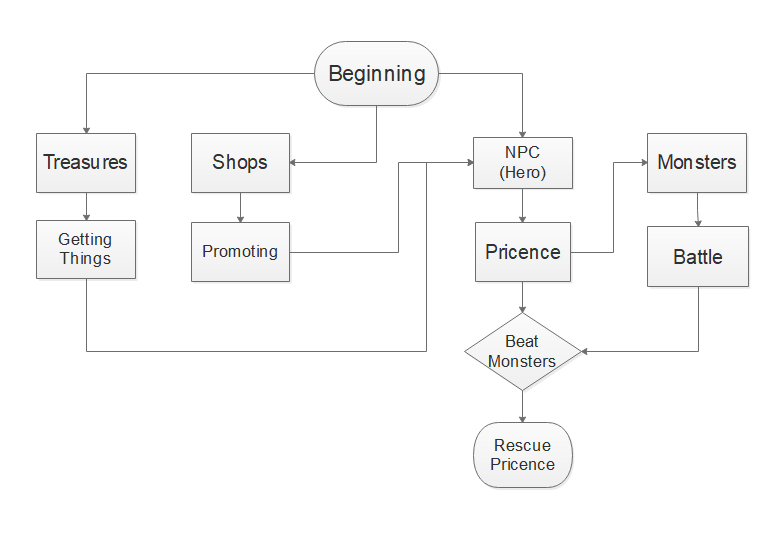
## 3.设计

3.1 设计思路原则

1.制作魔塔，我们首先需要有一个连贯完整的设计思路。也就是说制作时，我们要考虑到此作品给大家的特色是什么？我们希望大家走的路线有什么？诸如这些，都是要预先思考好的。比如：如果我们开发者希望玩家走攻击路线，那么就可以多设置一些攻高防低血量低的怪物。相反如果是均衡型，放置均匀的攻防和高血量的怪，往往玩家都会走防御路线。大忌是思路不连贯。如果你前面一堆坚固怪，造成前期难度太大，到最后期来一堆攻高防低血低的怪物，却没有跟上足够的资源发展英雄属性值，那么这样的玩家体验可想而知。制作魔塔，对游戏把握的要求很高。这虽然不是技术问题，但合理的设计思路，其重要性毋庸置疑。

2.  需要规避的枯燥设计  
①过多对称地图。没有迷惑性，路线明显，对称的地图一般都有这个缺点，如果想挑战高水准，那么这个往往是制作者的大忌。  
②过于堆积的怪物，往往会让玩家轻松发现此层的重点。路线更为直接、单一。这个就算是一个普通作者也要尽量避免。  
③左支右绌的钥匙周转   
高难度的作品往往突出于钥匙的周转。一条正确的线路往往是损失大但可保住基本的血量值而钥匙消耗最少的线路。所以，这条线路，往往要设置较强的怪物。而其他的路线都要放些低等怪物，这样一来，地图的难度和迷惑性就会大大增加。   
为此，尤其在自创地图上，我们需要自己反复测试，最终规划出一条合理的线路。然后设置在必经之路放置一些强怪。同时，在设计路线无关的门里，放一些中级怪物和合理补助资源，增加通关率。

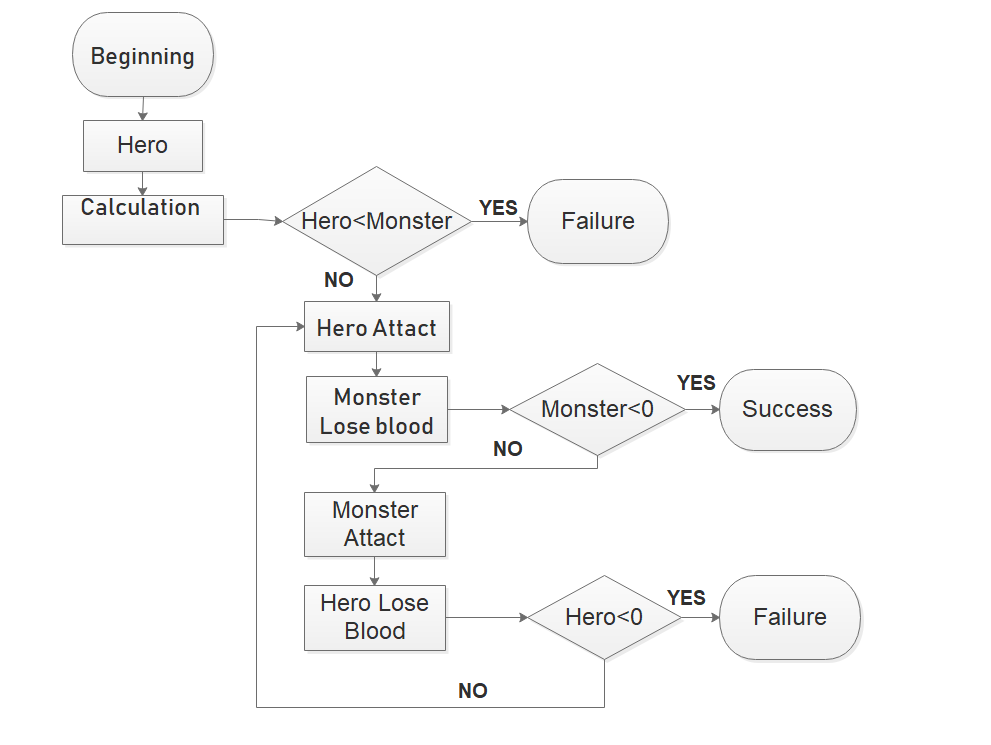
3.2 魔塔设计的总体逻辑



(Figure 1 Basic Logic)

3.3重点函数选例

3.3.1对战

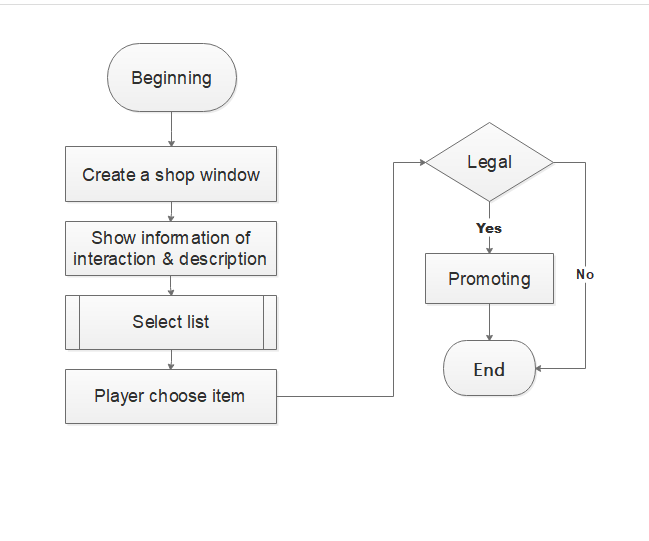


(Figure 2 Fighting Part)

如前所述，对战是我们更是玩家关注的重点内容，这里的细节会在后面第4板块展现。我们对对战是这样设计的，当玩家走进触发对战区域时将调用该函数。在激发对战之前会有一个实力比较，起初我们采用人与怪攻防值和进行比较但发现后面会导致更多繁杂判断，于是我们将攻防拆开来看，若玩家攻击值大于怪的防御值，则作差得到玩家伤害能力值，进行下一步，比较怪的攻击值和玩家的防御值，若怪物的攻击值大于玩家防御值则作差得到怪物的伤害能力值，否则怪的伤害能力值视为零并激发对战窗口；再下一步，各自用自己的生命值除以敌方伤害能力值得到获胜回合数，若玩家英雄获胜所需回合数小于怪物的，那么就激发对战窗口，并在对战窗口显示每回合的双方属性值变化（需要时间控制），最后将怪对玩家的伤害（减去相应血量值），给予玩家的经验和金币反映在信息栏，对战结束。前面会造成玩家英雄死亡的情况都不能激发对战窗口，相反，只会给出实力不足的友情提示。成功战胜怪物后执行穿越障碍的函数，类似于开门，把怪物的代号替换成代表地板的代号，并将该位置贴图替换成地板贴图。

其他细节参加实现板块。

3.3.2商店

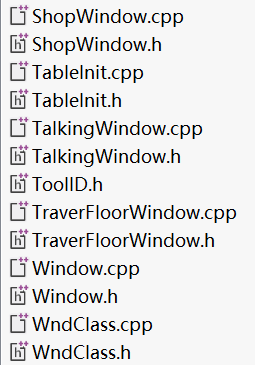
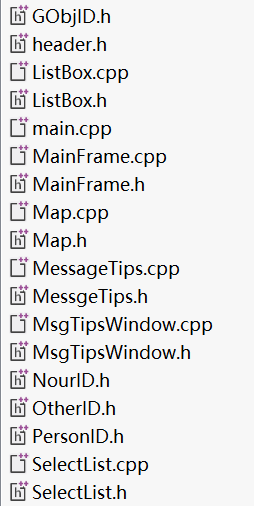
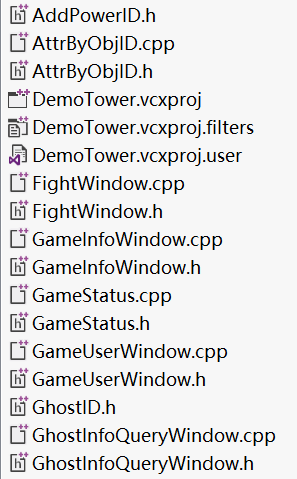


(Figure 3 Shop Part)

本魔塔在容易物资缺乏的关卡设置了商店，个别楼层会有两个商店，商店的“货币”有经验值、金币乃至钥匙，商品为属性值增益或钥匙。商店对通关也很重要。根据我们的设计，玩家只靠杀怪和捡道具是不可能打败魔王的，合理使用不同类型的商店可以令通关事半功倍。在算法上，商店的逻辑链条还是比较明了的，每做一次选择相当于做了一次循环，完成购买或取消购买即跳出该循环，如上图所示。其中，Select list 是我们专门为游戏中的选择操作编写的函数，合法性判断主要是判断玩家货币是否足以购买相应商品以及玩家中途是否选择退出放弃购买。

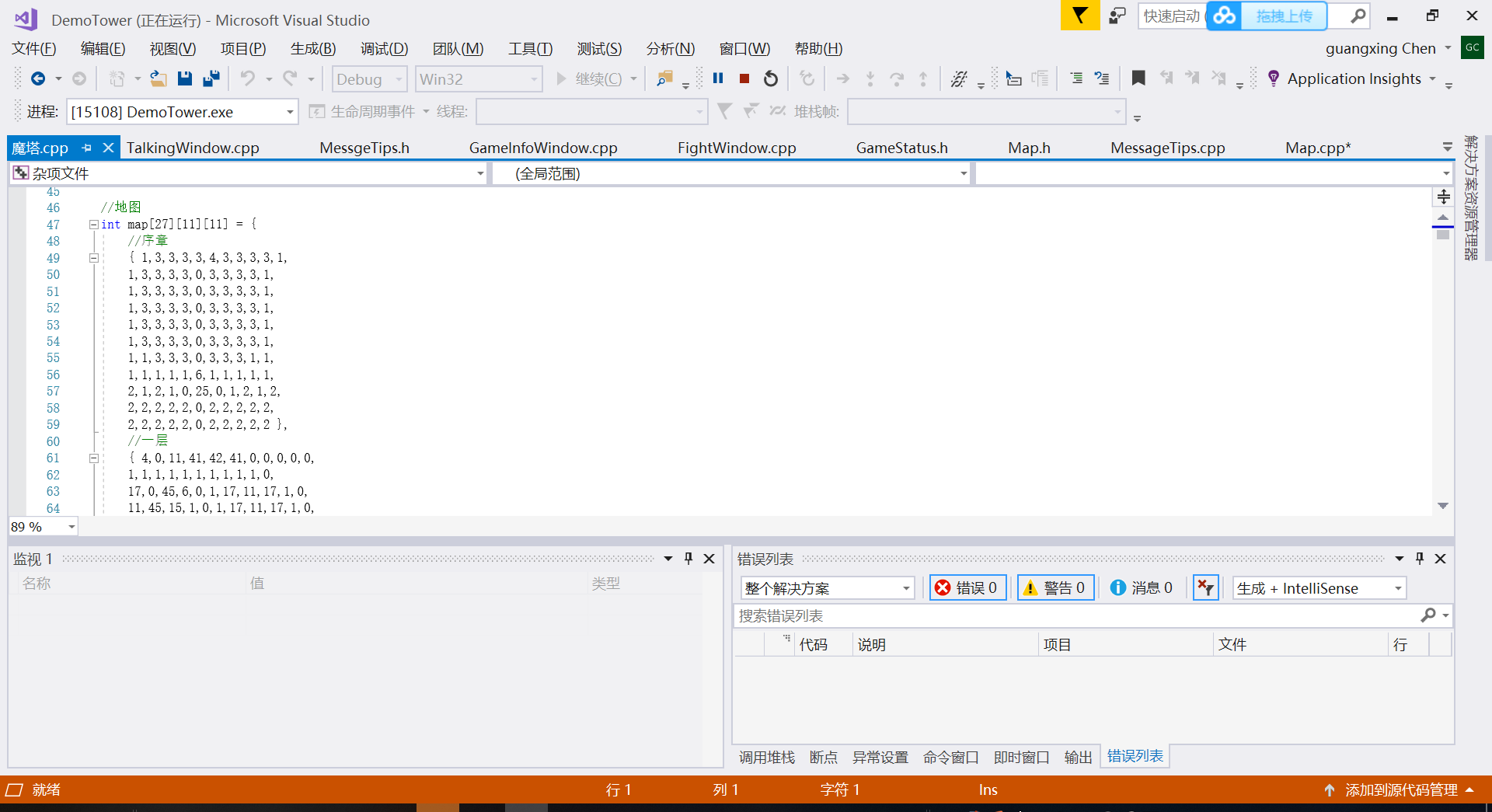
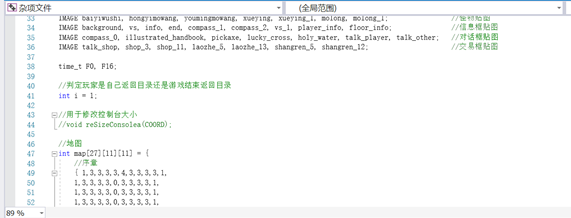
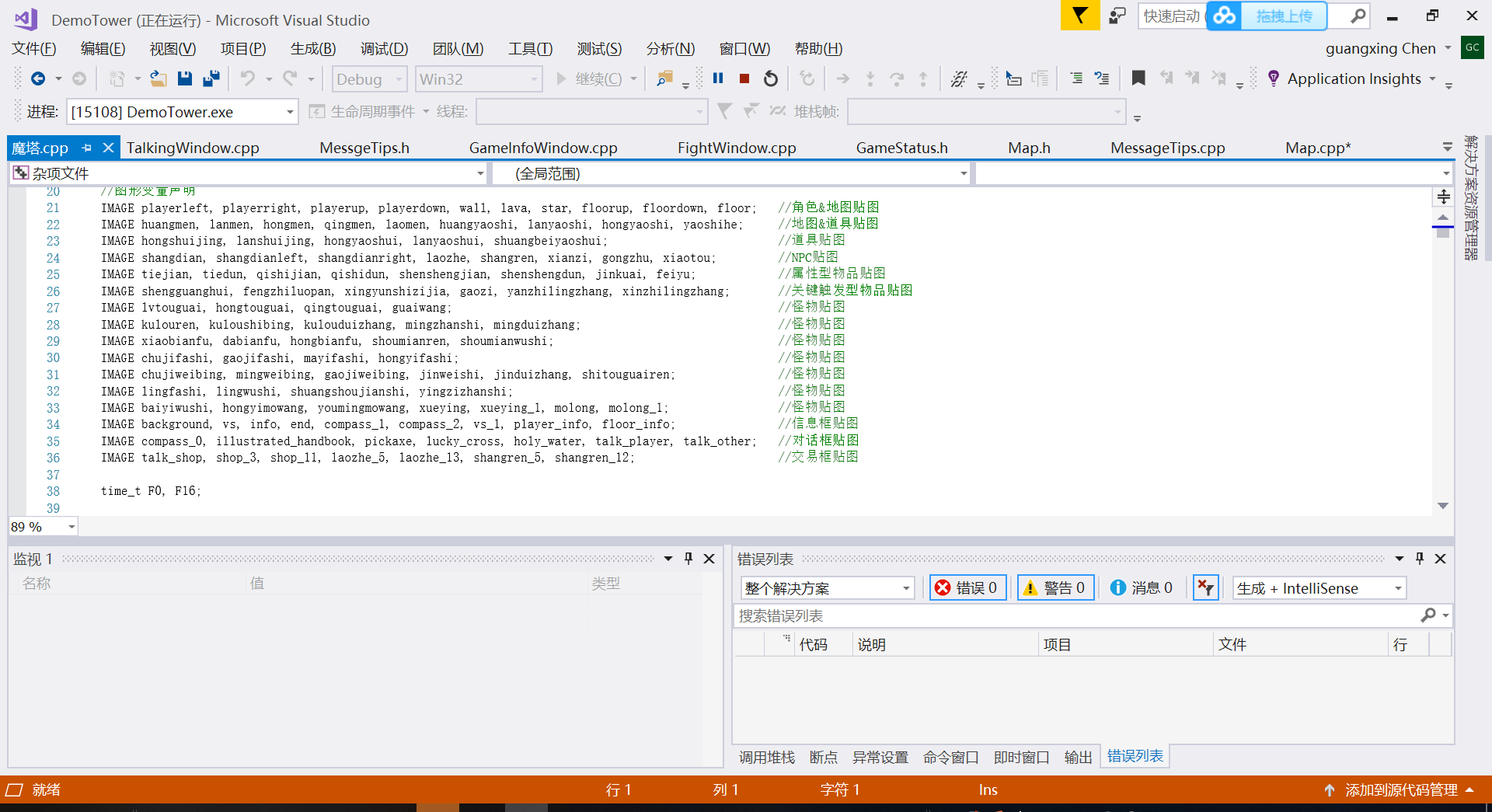
3.4程序函数一览

（Figure 4 All functions）



## 4实现

这里我们对前文问题陈述的内容进行回答。

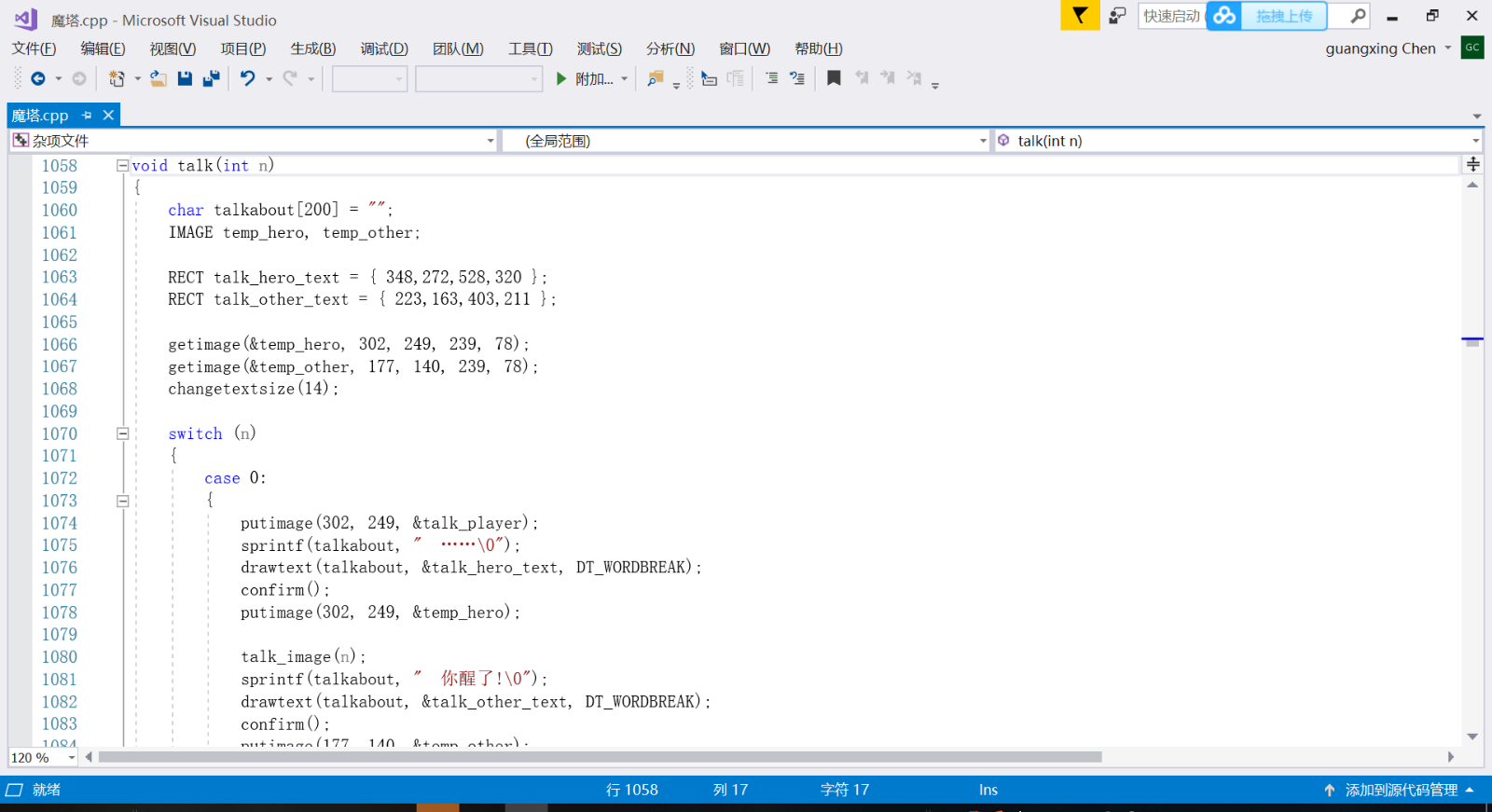
UI界面和地图的绘制是相通的。这里以地图为例谈代码实现。图形变量声明，引用资源库。计算图片应在的像素点，并将对应的数字代号填入地图矩阵中，使得生成的窗口中各个图片被放置在指定位置。图片的引入会让UI界面的艺术效果有质的飞越。



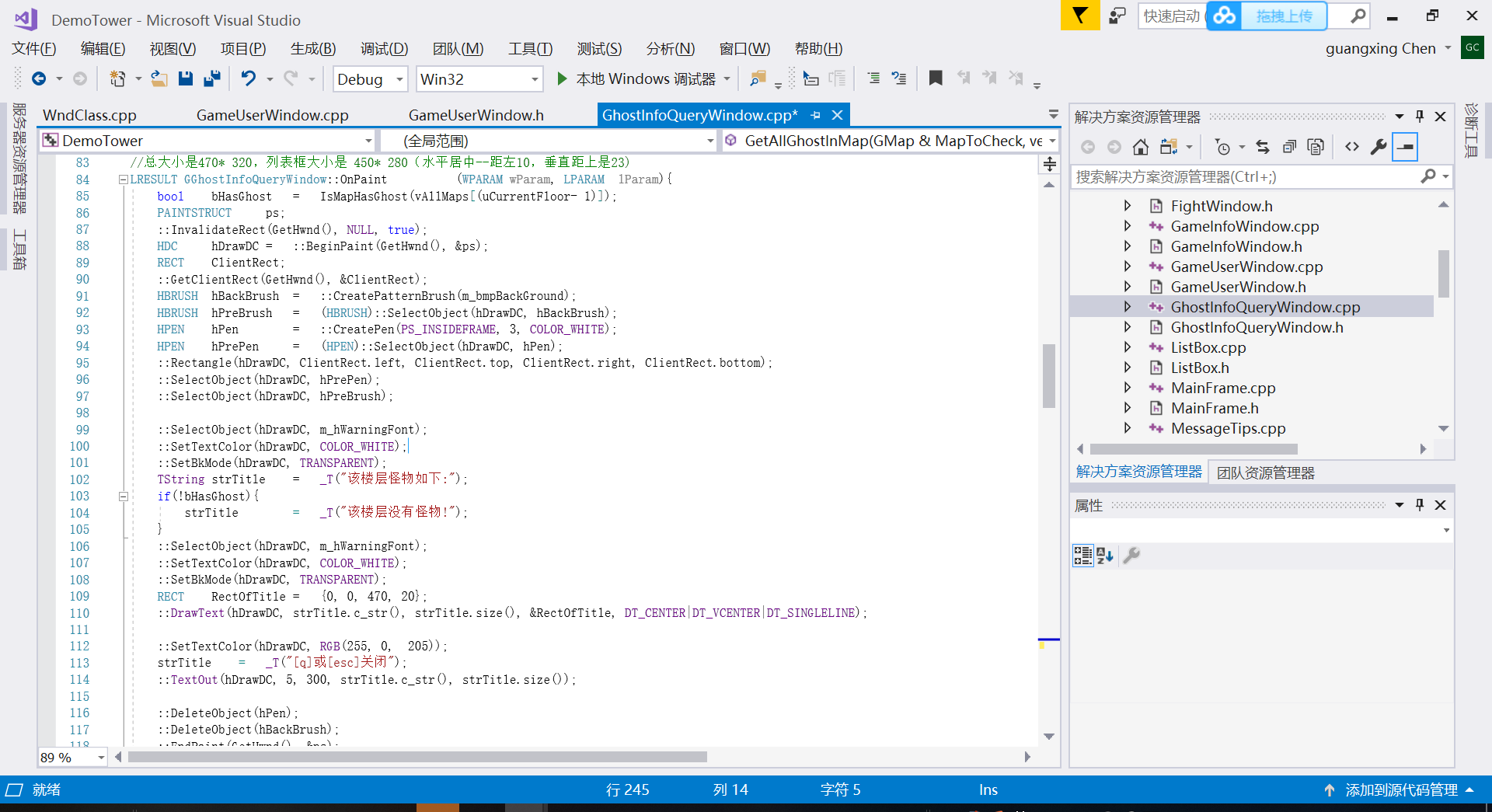
(Figure 5 Maps)

此外地图与地图之间我们建立了链接，激发区域即为梯子所在点。限于篇幅，这里从略。

英雄与NPC对话框,在刚开始编写代码时，我们未注意到要对窗口图片不变区域截图，结果导致刷新闪过于频繁，采用getimage函数对不变的地方截图并贴上窗口，只刷新对话文本，可实现预期效果。下图为解决方案二稿中的代码实现。终稿的代码实现见于程序中的Talking window,(终稿采用C++)。

  
（Figure 6 Talking window）

关于怪物和道具图鉴问题，我们特别设置了一个叫做“天眼”的道具，当玩家获得这个道具后按特定的键可以查看本楼层的怪物和道具信息，怪物、道具的注释和图片是一一对应的。这样一来虽无法做到鼠标移动至道具处显示道具注释，但也实现了目标功能，增加了趣味性。我们对快捷选择到达过的楼层也做了类似处理。下图为终稿中的实现办法（部分）



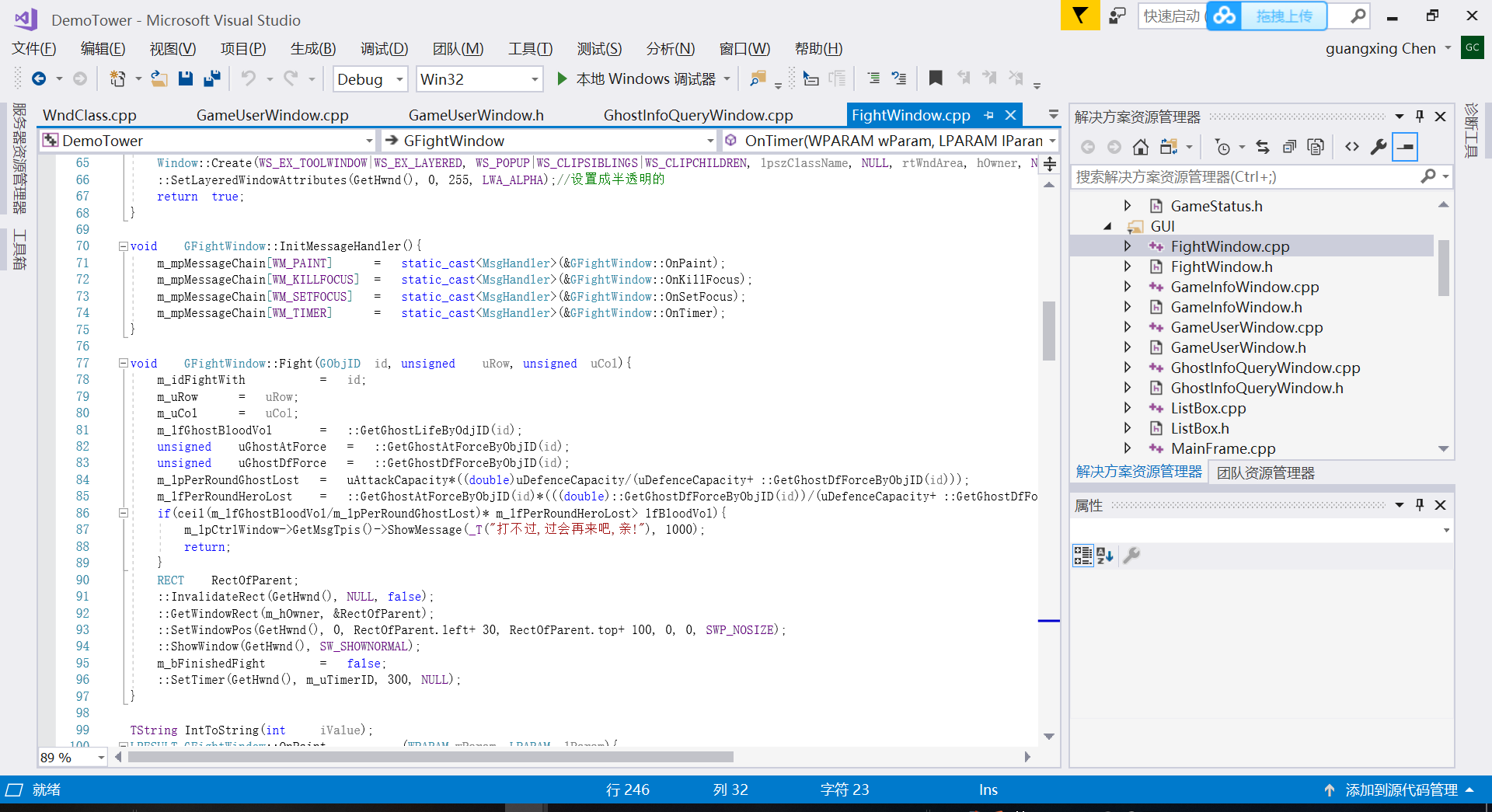
对战板块，我们删除了暴击设定，计算方法采用设计中所提到的办法。部分代码如下：

Figure7: Fighting Part

关于胜负的判定，当英雄消灭魔王后，游戏已经胜利。但我们决定不立即结束进程，而是任由玩家继续探索前面因为任务压力没有到达过的地方，给予玩家更多的自由，自己决定退出的时刻，这样也便于将制作者的所有设计展现在玩家面前。

而关于人物移动控制与商店菜单选择，我们实现了用方向键控制移动，用键盘和鼠标实现菜单选择。如商店菜单鼠标选择的一个函数如下：

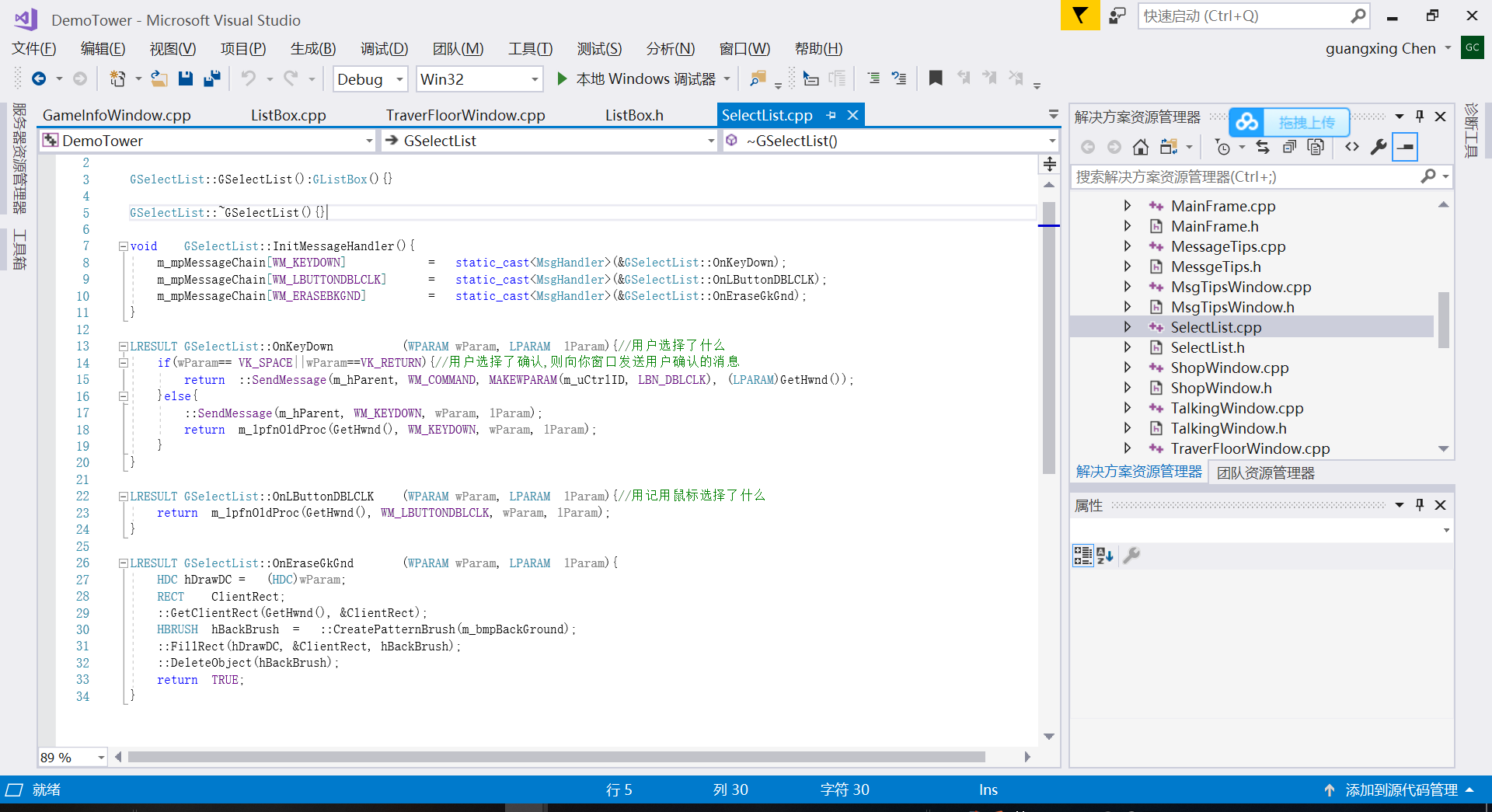


Figure8: Select list

## 5测试

|  |  |  |  |
| --- | --- | --- | --- |
| 调试项目 | 预期结果 | 实际结果 | 备注 |
| 不购买任何商品能否通关 | 不能通关 | 不能通关 | 无 |
| 键盘方向键控制人物移动情况是否正常 | 正常 | 正常 | 无 |
| 检查所有对话框运行情况 | 正常 | 闪屏现象严重 | 前文曾述该问题，已经更改代码，问题解决。 |
| 检查对战窗口三十次（分两种情况：实力不足不能触发对战，实力足够可以正常对战） | 正常 | 有三次运行错误 | 检查代码，发现按攻防和算的层进的判断有逻辑错误，当怪和英雄的属性值达到某种关系会恰好出现英雄扣血成负的非法情况，已经修改，细化了if的层次划分。 |
| 检查商店消费功能 | 正常 | 正常 | 无 |
| 检查特殊道具（重点：天眼、四叶草传送门） | 正常 | 正常，但天眼查看图鉴窗口过小，注释过密，影响玩家体验 | 删除了对魔法道具的注释，保留怪物图鉴。 |
| 检查怪物设置（种类、位置、数量、属性）是否合理 | 合理 | 第18关怪物强度过大，难度不合理 | 将高级怪换成中级怪，降低怪物的攻击，适当提高怪物的防御 |

## 结果&结论

通过老师学长学姐的指导以及小组成员的勠力合作，我们用C语言（应用了C++）完成了“魔塔”工程。该项目生动再现了魔塔这款经典游戏的诸多元素，易上手却也不乏策略挑战性。我们完成了构建魔塔游戏所用的函数并实现了魔塔的基本功能。可以说预期目标已经实现。我们把所学知识用到这项工程中，并克服了一些技术之外的一些困难，如地图设计、怪物参数调整过程中反反复复的测试，甚至在水平无奈之下放弃一些原创想法而改用稳妥的模板，技术上遇到的挫折也是不胜枚举，比如开始连地图都不会做，只能完成文字游戏，但好在最后出来的结果在总体上是成功的。本项目中除了在若干楼层的原创地图和路线外，终极任务完成后的自由探索，NPC交互语言以及特殊道具的设定（天眼图鉴、四叶草传送门）也是基础魔塔所不具备的亮点所在。在对战算法上作了一些改良，实力不足不会触发，对战对新手更加友好，不会因为挑战失败就从头再来。总之，希望这款小游戏能带给玩家闲暇时间的放松与欢愉。

尽管如此，限于实力和时间，我们也肯定有很多不足，比如当玩家耽于对战不知补给，结果血量、金钱和经验都不足以打败道路上的怪而往上前进，这是不可能通关的，然而我们的系统却没法识别出来做出判负。诸如此类的问题还望师长不吝批评指正。

感谢徐教授和AT教授长期以来的教诲，感谢学长学姐的关怀与帮助，感谢同学尤其是我的伙伴们的团结合作！